

Oracle® Database

Using Oracle GoldenGate on Oracle Cloud Marketplace



Release 19c (19.1.0.0)

F18630-14

October 2025



Copyright © 2019, 2021, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

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

Preface

Audience	i
Documentation Accessibility	i
Related Information	i
Conventions	i

Part I Oracle GoldenGate Microservices on Oracle Cloud Marketplace

1 Getting Started with Oracle Cloud Marketplace

Overview	1
Resources	1
Cross Cloud Functionality	2

Part II Using Oracle GoldenGate Microservices on Oracle Cloud Marketplace

2 Provisioning Oracle GoldenGate Microservices on Marketplace

Setting up Oracle GoldenGate Microservices for Your Database	1
Prerequisites to Deploy Oracle GoldenGate Microservices	1
Finding Oracle GoldenGate within the Oracle Cloud Marketplace	2
Deploying Oracle GoldenGate Microservices on Oracle Cloud Marketplace	2
Getting Started with Oracle GoldenGate Microservices	5
Configuring Source or Target Database for Replication	5
Changing Default Administrator Password	5
Using Service Manager	5
Using Administration Server	6
Creating User Accounts	6
Using Service Manager	7
Using Administration Server	7
Establishing Connectivity	7
Updating Network Related Files	8

Creating Database Credentials	9
Adding SchemaTrandata	10
Checking Tables with Added SchemaTrandata	10
Enabling Checkpoint Table	10
Implementing Heartbeat Monitoring	11
Configuring Capture Support	11
Configuring Apply Support	12
Connecting to Data Resources	13
Connecting to Oracle Database (on-premises)	13
Configure Oracle Database for Replication	13
Configure Oracle GoldenGate Microservices Compute Node	14
Architectures	14
Securely Connecting Oracle GoldenGate Microservices On Premise to Oracle GoldenGate Microservices on Marketplace	14
Connecting Through Public IP Address	15
Connecting to Oracle Database as a Service (DBaaS)	18
Configure Oracle Database for Replication	19
Configure Oracle GoldenGate Microservices Compute Node	19
Connecting to Oracle Autonomous Data Warehouse/Autonomous Transaction Processing	20
Configure the Autonomous Data Warehouse for Replication	20
Autonomous Database Client Credentials	21
Configure Oracle Goldengate Microservices for Replication	23

Part III Using Oracle GoldenGate Classic on Oracle Cloud Marketplace

3 Oracle GoldenGate Classic for Oracle

Deploying Oracle GoldenGate Classic Architecture for Oracle Database	1
Finding Oracle GoldenGate within the Oracle Cloud Marketplace	1
Launching Oracle GoldenGate Classic	1
Getting Started with Oracle GoldenGate Classic Compute Node	2
Running GGSCI	3
Creating Oracle GoldenGate Subdirectories	3
Getting Started with Oracle GoldenGate Classic for Oracle Database	4
Configuring Source or Target Database for Replication	4
Establishing Connectivity with Oracle GoldenGate Classic	4
Updating Network Related Files for Oracle GoldenGate Classic	5
Creating Database Credentials for Oracle GoldenGate Classic	6
Adding SchemaTrandata for Oracle GoldenGate Classic	6
Creating Checkpoint Table for Oracle GoldenGate Classic	7
Enabling Heartbeat Table for Oracle GoldenGate Classic	7

Configuring Capture Support for Oracle GoldenGate Classic	8
Configuring Apply Support for Oracle GoldenGate Classic	9
Connecting to Data Resources for Oracle GoldenGate Classic	10
Connecting to Oracle Database (on-premises) for Oracle GoldenGate Classic	10
Configure Oracle Database for Replication	10
Configure Oracle GoldenGate Classic Compute Node	11
Connecting to Oracle Database as a Service for Oracle GoldenGate Classic	11
Configure Oracle Database for Replication	12
Configure Oracle GoldenGate Classic Compute Node	12
Connecting to Oracle Autonomous Data Warehouse/Autonomous Transaction Processing	13
Configure the Autonomous Data Warehouse for Replication	13
Autonomous Database Client Credentials	14
Configure Oracle Goldengate Classic for Replication	16

4 Oracle GoldenGate Classic for DB2 for z/OS

Deploying Oracle GoldenGate Classic Architecture for DB2 for z/OS Platforms	1
Finding Oracle GoldenGate for Non-Oracle (DB2 for z/OS) within the Oracle Cloud Marketplace	1
Launching Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS)	1
Getting Started with Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) Compute Node	3
Installing DB2 drivers	3
Running GGSCI	3
Working with Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS)	4
Configuring Source and Target Database for Replication	4
Establishing Connectivity	4
Configuring Capture Support	5
Configuring Apply Support	5
Connecting to Data Resources	6
Connecting to DB2 z/OS (On-Premises)	6
Configuring Oracle DB2 z/OS for Replication	6
Configuring Oracle GoldenGate Classic for Non-Oracle (DB2 z/OS) Compute Node	6

5 Oracle GoldenGate Classic for Big Data

Deploying Oracle GoldenGate Classic Architecture for Big Data and Cloud Platforms	1
Finding Oracle GoldenGate for Big Data within the Oracle Cloud Marketplace	1
Launching Oracle GoldenGate Classic for Big Data	1
Getting Started with Oracle GoldenGate Classic for Big Data Compute Node	3
Installing Third Party Drivers or Libraries	3
Running GGSCI	3

Working with Oracle GoldenGate Classic for Big Data	4
Configuring Source and Target Data System for Replication	4
Establishing Connectivity	4
Configuring Capture Support	4
Configuring Apply Support	5
Connecting to Data Resources	6
Connecting to Big Data Systems	6
Configuring Big Data Systems for Replication	6
Configuring Oracle GoldenGate Classic for Big Data Compute Node	7

6 Oracle GoldenGate Classic for SQL Server

Deploying Oracle GoldenGate Classic Architecture for SQL Server Platforms	1
Finding Oracle GoldenGate for Non-Oracle (SQL Server) within the Oracle Cloud Marketplace	1
Launching Oracle GoldenGate Classic for Non-Oracle (SQL Server)	1
Connecting to Oracle GoldenGate Classic for Non-Oracle (SQL Server) Compute Node	3
Working with Oracle GoldenGate Classic for Non-Oracle (SQL Server)	3
Establishing Connectivity	4
Oracle GoldenGate for SQL Server Requirements	4
Creating Database User and Schema	4
Installing Microsoft ODBC Drivers for Linux	5
Configuring ODBC Connectivity	5
Creating GLOBALS File and Starting GGSCI	6
Enabling Supplemental Logging for a Source SQL Server Database	6
Configuring and Creating a Remote Extract for SQL Server	7
Configuring and Creating a Remote Replicat for SQL Server	8

7 Oracle GoldenGate Classic for PostgreSQL

Deploying Oracle GoldenGate Classic Architecture for PostgreSQL Platforms	1
Finding Oracle GoldenGate for Non-Oracle (PostgreSQL) within the Oracle Cloud Marketplace	1
Launching Oracle GoldenGate Classic for Non-Oracle (PostgreSQL)	1
Connecting to Oracle GoldenGate Classic for Non-Oracle (PostgreSQL) Compute Node	3
Working with Oracle GoldenGate Classic for Non-Oracle (PostgreSQL)	3
Establishing Connectivity	4
Oracle GoldenGate for PostgreSQL Requirements	4
Creating Database User and Schema	5
Setting up and Configuring odbc.ini File	5
Starting GGSCI	5
Registering a Replication Slot	5
Enabling Supplemental Logging for a Source PostgreSQL Database	6

Configuring and Creating a Remote Extract for PostgreSQL	6
Configuring and Creating a Remote Replicat for PostgreSQL	7

8 Oracle GoldenGate Classic for MySQL

Deploying Oracle GoldenGate Classic Architecture for MySQL Platforms	1
Finding Oracle GoldenGate for Non-Oracle (MySQL) within the Oracle Cloud Marketplace	1
Launching Oracle GoldenGate Classic for Non-Oracle (MySQL)	1
Connecting to Oracle GoldenGate Classic for Non-Oracle (MySQL) Compute Node	3
Working with Oracle GoldenGate Classic for Non-Oracle (MySQL)	3
Establishing Connectivity	4
Oracle GoldenGate for MySQL Requirements	4
Configure Source MySQL Database for Remote Capture	4
Configure Target MySQL Database for Remote Apply	5
Starting GGSCI	5
Configuring and Creating a Remote Extract for MySQL	5
Configuring and Creating a Remote Replicat for MySQL	6

Part IV Additional Topics

9 Upscaling and Downscaling Oracle GoldenGate Marketplace Compute Shape Sizes

10 Migrating Oracle GoldenGate Cloud Service (Gen 1) DIPC to Oracle GoldenGate Classic on Oracle Cloud Marketplace

11 Upgrading Oracle GoldenGate Marketplace Image

Stacks	1
Retrieving a Stack	1
Identifying the Terraform Version for the Stack to be Upgraded	1
Upgrading an Oracle Cloud Marketplace Image (To Different Versions of Terraform)	2
Verify that the Stack you want to Upgrade is up to Date	2
Download the Terraform Configuration and Terraform State File from the Old Stack and Create a New Stack	2
Upgrade from Terraform v0.12 to v1.0.x	3
Upgrading an Oracle Cloud Marketplace Image to 1.5.x (Same Terraform Version)	4

12 Removing Oracle GoldenGate Instance

13 OCI Policies Needed for Creating Oracle GoldenGate Marketplace Instances

14 Troubleshooting

Logdump	1
Enable Debug Logging for Oracle GoldenGate Microservices	1

Preface

This guide describes how to use Oracle GoldenGate on Oracle Cloud Marketplace.

Audience

This guide is intended for the person or persons who are responsible for operating Oracle GoldenGate and maintaining its performance. This audience typically includes, but is not limited to, systems administrators and database administrators.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Information

The Oracle GoldenGate Product Documentation Libraries are found at

<https://docs.oracle.com/en/middleware/goldengate/index.html>

For additional information on Oracle GoldenGate, refer to, <https://www.oracle.com/middleware/technologies/goldengate.html>

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, such as "From the File menu, select Save ." Boldface also is used for terms defined in text or in the glossary.
<i>italic</i>	Italic type indicates placeholder variables for which you supply particular values, such as in the parameter statement: TABLE <i>table_name</i> . Italic type also is used for book titles and emphasis.
<i>italic</i>	

Convention	Meaning
monospace	Monospace type indicates code components such as user exits and scripts; the names of files and database objects; URL paths; and input and output text that appears on the screen. Uppercase monospace type is generally used to represent the names of Oracle GoldenGate parameters, commands, and user-configurable functions, as well as SQL commands and keywords.
UPPERCASE	Uppercase in the regular text font indicates the name of a utility unless the name is intended to be a specific case.
{ }	Braces within syntax enclose a set of options that are separated by pipe symbols, one of which must be selected, for example: { <i>option1</i> <i>option2</i> <i>option3</i> }.
[]	Brackets within syntax indicate an optional element. For example in this syntax, the SAVE clause is optional: CLEANUP REPLICAT <i>group_name</i> [, SAVE <i>count</i>]. Multiple options within an optional element are separated by a pipe symbol, for example: [<i>option1</i> <i>option2</i>].

Part I

Oracle GoldenGate Microservices on Oracle Cloud Marketplace

You need to use the *Oracle GoldenGate Microservices Documentation* along with the *Using Oracle GoldenGate on Oracle Cloud Marketplace* guide to use Microservices on Marketplace.

Prerequisites

Ensure that you go through the following prerequisites before using Oracle GoldenGate on Oracle Cloud Marketplace.

Supported Browsers

Oracle Cloud Infrastructure supports the latest desktop versions of Google Chrome, Microsoft Edge, Internet Explorer 11, Safari, Firefox, and Firefox ESR. Note that Mobile browsers as well as private browsing mode is not supported for Firefox, Internet Explorer, or Edge.

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 Generating public/private rsa key pair.
```

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*.

Getting Started with Oracle Cloud Marketplace

This chapter provides an introduction to Oracle GoldenGate on the Oracle Cloud Marketplace.

Topics:

- [Overview](#)
- [Resources](#)
- [Cross Cloud Functionality](#)

Overview

Oracle GoldenGate on Oracle Cloud Marketplace provides pre-installed versions of Oracle GoldenGate and offers the same benefits, scalability, security, and robustness that you can rely on for your enterprise replication needs.

Resources

The Oracle GoldenGate image on Oracle Cloud Marketplace contains the latest Oracle GoldenGate release along with Oracle Database Client software for all supported versions of Oracle Database.

For supported Oracle Database platforms for Oracle GoldenGate, see the [latest certification matrix](#).

All the software on Oracle GoldenGate Compute node is installed under the `/u01/app` directory structure.

Supported Compute Shapes

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

Compute Shape	CPU	Memory	Bandwidth
VM.Standard2.4	4	60	4.1 Gbps
VM.Standard2.8	8	160	8.2 Gbps
VM.Standard2.16	16	240	16.4 Gbps
VM.Standard2.24	24	320	24.6 Gbps

For pricing details on compute nodes, please refer to [OCI Pricing](#).

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	50GB	Yes
Swap	256GB	Yes
Trail	512GB	Yes
Deployments	128GB	Yes
Cache Manager	128GB	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.

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
Google Cloud	Yes	Yes

Part II

Using Oracle GoldenGate Microservices on Oracle Cloud Marketplace

This part helps you to get started with Oracle GoldenGate Microservices on Oracle Cloud Marketplace.

This part contains the following chapters:

- [Provisioning Oracle GoldenGate Microservices on Marketplace](#)

Provisioning Oracle GoldenGate Microservices on Marketplace

This chapter describes the steps to provision Oracle GoldenGate Microservices for Oracle Cloud Marketplace.

Setting up Oracle GoldenGate Microservices for Your Database

Before you can start using Oracle GoldenGate Microservices, there are a few tasks that you must perform, depending on your database platform, to ensure that your environment is complete and ready to replicate your data.

Prerequisites to Deploy Oracle GoldenGate Microservices

Here are the prerequisites required to deploy Oracle GoldenGate Microservices:

- [Oracle Cloud Account](#)
- Access to assigned Oracle Cloud Tenant
- Compute node resources within 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.

Perform the following steps to build your SSH keys:

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

```
$ ssh-keygen Generating public/private rsa key pair.
```

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*.

Finding Oracle GoldenGate within the Oracle Cloud Marketplace

Follow the below steps to launch Oracle GoldenGate Microservices from Oracle Cloud Marketplace :

1. Log in to [Oracle Cloud Marketplace](#).
2. From the Oracle Cloud Marketplace home page, use the search box under Applications and search for the keyword GoldenGate.
3. From the search results, select Oracle GoldenGate.

Deploying Oracle GoldenGate Microservices on Oracle Cloud Marketplace

After finding Oracle GoldenGate Microservices listing in Oracle Cloud Marketplace, you can deploy Oracle GoldenGate using the provided Stack Listing. This TerraForm Stack prompts you for specific information and then builds the Oracle Cloud Infrastructure Compute Node with the desired hardware settings, Oracle Database Client, Oracle GoldenGate Microservices and up to two default deployments.

Once you have found Oracle GoldenGate from the search results on Oracle Cloud Marketplace, follow the below steps to deploy Oracle GoldenGate Microservices on Oracle Cloud Marketplace using the Stack Listing.

1. From the Application page, select **Get App** to display the option to select either **Commercial Market** or **Government Market**.
2. If you select Government Market, then select **OCI Region** from the drop-down menu.
3. Click **Sign In** and enter the OCI tenant details.
4. Sign in to the Identity provider.
5. On the **Oracle GoldenGate for Oracle** page, provide the following information:
 - **Select Version** - It provides a list of versions that are available in the listing. Select the required release version.
 - **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** - This check box is selected by default. Oracle recommends to review the licenses before proceeding with the instance creation.
 - **Launch Stack** - It launches the stack in the OCI environment.
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 that you provide while creating the Stack.
- Create In Compartment – It defaults to the compartment you have selected on the Oracle GoldenGate 19c for Oracle page.
- Terraform Version - It defaults to the default version supported.
- 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 compute node with Oracle GoldenGate Microservices.

- Name for New Resources -
 - Display Name** – Display name used to identify all new OCI resources.
 - Host DNS Name** – Name of the Domain Name Service for the new compute node.
- Network Settings -
 - 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.
 - Network Compartment** (optional) – Compartment for new or existing network resources.
 - VCN** (optional) – Existing VCN to use for the newly created instance if you are not creating a new network.
 - Subnet Network Compartment** (optional) - It specifies the compartment in which the VCN submit resides.
 - Subnet** (optional) – Existing subnet to use for the newly created instance if you are not creating a new network. The subnet that you have selected must match the same Availability Domain set in the Instance Settings.
- Instance Settings -
 - Availability Domain** – It specifies the availability domain for the newly created Oracle GoldenGate Instance. It must match the Subnet that you have selected in the Use Existing Network settings.
 - Compute Shape** – Shape of new compute instance. Supported shapes are VM.Standard2.4, VM.Standard2.8, VM.Standard2.16 and VM.Standard2.24.
 - Assign Public IP** – This option indicates if the newly created VM should have a public IP address. This option is selected by default. If you clear this check box, no public IP address will be assigned preventing public access to the compute node.

ⓘ 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.

- d. **Custom Volume Sizes**- Select this check box to customize the size of the new block storage volumes that are built for the compute node.
Block Storage (Custom Volume Sizes) -
 - i. Boot Volume Size – Default value is 50GB
 - ii. Swap Volume Size – Default value is 256GB
 - iii. Trails Volume Size – Default value is 512GB
 - iv. Deployments Volume Size – Default value is 128GB
- **Create OGG Deployments -**
 - a. **Deployment 1 – Name** (mandatory) – Name of the first Oracle GoldenGate Microservices deployment.
 - b. **Deployment 1 – Database** (mandatory) – Oracle Database version for deployment 1. Supported Oracle Database versions are:
 - Oracle 11g – Used for Oracle Database 11.2.0.4
 - Oracle 12c – Used for Oracle Database 12.1.x and 12.2.x
 - Oracle 18c – Used for Oracle Database 18.x
 - Oracle 19c – Used for Oracle Database 19.x
 - c. **Deployment 2 – Name** (optional) – Name of the second Oracle GoldenGate deployment
 - d. **Deployment 2 – Database** (optional) – Oracle Database version for deployment 2. Supported Oracle Database versions are:
 - Oracle 11g – Used for Oracle Database 11.2.0.4
 - Oracle 12c – Used for Oracle Database 12.1.x and 12.2.x
 - Oracle 18c – Used for Oracle Database 18.x
 - Oracle 19c – Used for Oracle Database 19.x
 - e. **Deployment 2 - Autonomous Database Compartment** (optional) - Select this option if deployment 2 replicates to an Autonomous Database Warehouse. By selecting this option, the Autonomous Database Warehouse wallets and credentials get imported thereby making it easier to connect to the Autonomous Database Warehouse.
 - Deployment 2 - Autonomous Database Compartment - Specifies the compartment in which existing Autonomous Database resides.
 - Deployment 2 - Autonomous Database Instance - Choose an Autonomous Database Instance.
- **Shell Access -**
 - **SSH Public Key** - Public Key for allowing SSH access as the `opc` user. Enter the key and click **Next**.

8. On the **Review** page, review the information you provided and then click **Create**.

9. After clicking **Create**, you are navigated to the **Stacks Job Details** page. You can monitor the creation of the compute node using this page.
10. Upon completion, you can now view the Oracle GoldenGate Microservices compute node under **Instances**.

Getting Started with Oracle GoldenGate Microservices

After deploying Oracle GoldenGate Microservices on the Oracle Cloud Marketplace, you can access the latest release of Oracle GoldenGate.

Configuring Source or Target Database for Replication

Before you can begin replicating, you should prepare the source or target database to support Oracle GoldenGate. To prepare your database, follow the steps listed in [Preparing the Database for Oracle GoldenGate](#) section of *Using Oracle GoldenGate for Oracle Database* guide. The steps listed in the *Using Oracle GoldenGate for Oracle Database* guide helps you to enable logging and kernel parameters, set the flashback query and manage server resources.

Changing Default Administrator Password

Changing passwords for critical accounts, such as oggadmin, is the first priority in securing your Oracle GoldenGate Microservices deployment. To change the password for oggadmin, you must first change it in both Service Manager and Administration Server. The following sections guide you to do this with Service Manager and Administration Server.

- [Using Service Manager](#)
- [Using Administration Server](#)

Using Service Manager

After logging into the Oracle GoldenGate Microservices Service Manager as the Administrator for the deployments, you have to change the password for the Security Role user. In order to do this, perform the following steps:

1. Navigate to the Service Manager login page. You can reach the Service Manager page by using the public IP address that you obtained when you performed a look up of the compute node information for the environment.

`https://<public_ip_address>`

2. Log in using the oggadmin user and the password credentials located in the `/home/opc/ogg-credentials.json` file.
3. Once you have logged into the Service Manager, use the menu icon present on the top left corner to open the menu.
4. Select Administrator option from the menu.
5. From the Users screen, select the pencil icon under Action option.
6. Update the essential details for password and info sections for the oggadmin user and click Submit.
7. Upon successful reset, the current user will be logged out. Log in again to the Service Manager by using the new password.

ⓘ Note

Passwords must be 8 to 30 characters long and must contain at least 1 uppercase, 1 lowercase, 1 numeric, and 1 special character. Special characters such as '\$', '^', or '?' are not allowed.

Using Administration Server

After changing the Oracle GoldenGate Microservices Service Manager security role user password, you need to change the password of the security role user in the underlying deployments. In order to do this, perform the following steps:

1. From the Service Manager page, select the port number for the Administration Server in the deployment. This navigates you to the login page for that deployment.
2. Log in using the oggadmin user credentials available in the following location:

```
/home/opc/ogg_credentials.json
```

3. Post log in, use the menu icon present in the top left corner to open the menu section.
4. From the menu, select the Administrator option.
5. From the Users screen, select the pencil icon under Action option.
6. Update the essential details in the password and info sections for the oggadmin user and click Submit.
7. After successfully resetting the password, the current user will be logged out. Log in again to the Administration Server using the new password.

 ⓘ Note

Passwords must be 8 to 30 characters long and contain at least 1 uppercase, 1 lowercase, 1 numeric, and 1 special character. Special characters such as '\$', '^', or '?' are not allowed.

Creating User Accounts

In order to secure your Oracle GoldenGate Microservices deployment, you have to add user accounts for Oracle GoldenGate Users. Oracle GoldenGate Users should be assigned privileges based on functional roles that they are expected to perform. These roles are:

- Security
- Administrator
- Operator
- User

Oracle GoldenGate users only have access permissions according to their defined access levels. For more information on how Oracle GoldenGate Security Framework is used, refer to [Securing Oracle GoldenGate](#) guide.

Topics:

- [Using Service Manager](#)
- [Using Administration Server](#)

Using Service Manager

After logging into the Oracle GoldenGate Microservices Service Manager as the administrator for the deployments, you have to create a new user with the role of Administrator, Operator, or User to administer the architecture. Users with the security role can administer the entire architecture.

1. Navigate to the Service Manager login page. You can reach the Service Manager page by using the public IP address that you obtained when you looked up the compute node information for the environment.

`https://<public_ip_address>`

2. Log in using the `oggadmin` user and the password credentials located in the `/home/opc/ogg_credentials.json` file.
3. Click the menu icon present on the top left corner to open the menu section.
4. Select the Administrator option from the menu.
5. On the Users screen, select the plus (+) icon to add a new user.
6. Fill in all the required fields.
7. Click Submit to create the new user.

Using Administration Server

After logging into the Oracle GoldenGate Microservices Administration Server as the Administrator, for the specified deployment, you have to create a new user with the role of Administrator, Operator, or User to administer the deployment. To do this, perform the following steps:

1. Log in to the Administration Service using the Security Role User (`oggadmin`) credentials.
2. After logging in to the Administration Service, click the menu icon present in the top left corner to open the menu.
3. From the menu, select the Administrator option.
4. From the Users screen, select the plus (+) icon, to create a new user.
5. Fill in the details for all the required fields and click Submit.

Note

Passwords must be 8 to 30 characters long and contain at least 1 uppercase, 1 lowercase, 1 numeric, and 1 special character. Special characters such as '\$', '^', or '?' are not allowed.

Establishing Connectivity

The Oracle GoldenGate Microservice on Marketplace compute node acts as a hub where you can manage your connections to source and target database. To do this, establish a network

connection between the compute node and your source and target database. The compute node is pre-configured with the Oracle Database Client software and is ready to use.

In most cases, you have to provide `sqlnet.ora` and a `tnsnames.ora` file in the `TNS_ADMIN` directory, to be able to establish connection between source or target database and the compute node. On a deployment basis, the `TNS_ADMIN` directory has been established as `/u02/deployments/<deployment>/etc`.

To ensure network connectivity from the deployment, you have to set up certain additional things in the deployment home. Oracle recommends the location to be `/u02/deployments/<deployment>/etc`. To create this, as per Oracle Client release, perform the following steps.

1. Connect to the Oracle GoldenGate Marketplace Compute Node as the `opc` user.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Navigate to `/u02/deployments/<deployment>/etc`

```
$ cd /u02/deployments/<deployment>/etc
```

3. Copy the existing files `sqlnet.ora` and `tnsnames.ora` to the `TNS_ADMIN` directory. Add or update these two files on the compute node.

Note

- If you are using Oracle Autonomous products, the `tnsnames.ora` file is included in your `Client_Credentials.zip` file. You have to edit this `tnsnames.ora` file.
- If you want your networking directory structure to be consistent with other Oracle products, you must append `/network/admin` to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

Updating Network Related Files

The Oracle GoldenGate on Marketplace Compute Node comes pre-configured with Oracle Client installed. In order to establish network configuration between the compute node and the source or target systems within your architecture, you have to add or update the network related files in the Oracle Client.

You can find these files in the `TNS_ADMIN` location and the recommended location is `/u02/deployments/<deployment>/etc`. You have to add or update the network files, such as `sqlnet.ora` and `tnsnames.ora` on the compute node.

To update the files,

1. Connect to the Oracle GoldenGate Marketplace Compute Node, using `opc` user credentials.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Change directories to the location /u02/deployments/<deployment>/etc .

```
$ cd /u02/deployments/<deployment>/etc
```

3. Edit sqlnet.ora and tnsnames.ora files.

 **Note**

If you want the networking directory structure to be consistent with other Oracle products, you must append /network/admin to the directory structure. For this you have to change the environment variable TNS_ADMIN. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

4. In sqlnet.ora, comment out the following lines using the hash (#) character. Ensure to comment out these lines if they are connecting to a database outside the oracle (ora.com) domain.

```
#NAMES.DIRECTORY_PATH= (TNSNAMES, ONAMES, HOSTNAME)  
#NAMES.DEFAULT_DOMAIN = ora.com
```

The following is an example of a connection within the tnsnames.ora file:

```
TNS1 =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP) (HOST = 10.10.10.1) (PORT = 1521))  
    (CONNECT_DATA = (SERVER = DEDICATED)(SERVICE_NAME = TNS1))  
  )
```

For more information, see [Database Net Services Reference](#).

Creating Database Credentials

Use the credential store to store and use database credentials for the source and target databases for Oracle GoldenGate Microservices. Use the tnsnames.ora file to connect to the required database.

To create database credentials, perform the following tasks:

1. Log in to the Administration Server and configure the database credentials.
2. Open the context menu in the top left corner of the Overview page.
3. From the context menu, select Configuration.
4. From the Database tab, click the plus (+) icon to add a new credential.
5. Provide the following information and click Submit:
 - Credential Domain: [Defaults to OracleGoldenGate]
 - Credential Alias: [Name of the Alias]
 - User ID: ggadmin@<tnsnames_reference>
 - Password: [Password for ggadmin]
 - Verify Password: [Password for ggadmin]

6. Test the connection to the database by clicking on the database icon, after adding the credential.

Adding SchemaTrandata

After adding the credential for connecting to the source database, you must enable supplemental logging on the source schema. The following steps are used to add SchemaTrandata to the source schema:

1. Log in to the ServiceManager console.
2. From the ServiceManager main page, select the hyperlink for the port number associated with the Administration Service.
3. Open the Context menu present on the top left corner of the Overview page and from the Context menu select Configuration.
4. From the Database tab, select the database icon to log the source user to the database.
5. From the Transaction Information, select the plus (+) icon.
6. Ensure that you select the Schema option, provided the selected Schema Name should have schema trandata added.
7. Click Submit.

Note

If your source database is multitenant and you are connecting to the root using a c## user, from Oracle Database 12.1 and later, you have to specify the PDB database with the schema. i.e. < pdb > . < schema >.

Checking Tables with Added SchemaTrandata

After adding transactional data to a schema, you must validate the tables. You can do this by following the below steps:

1. Log in to the ServiceManager console.
2. From the ServiceManager main page, select the hyperlink for the port number associated with the Administration Service.
3. Open the Context menu present on the top left corner of the Overview page and from the Context menu, select Configuration.
4. From the Database tab, select the database icon to log the source user to the database.
5. From the Transaction Information, select the search icon.
6. Ensure that you select the Schema option and provide the Schema Name that you need to verify.
7. Click the search icon.

Enabling Checkpoint Table

Checkpoint tables contain the data necessary for tracking the progress of the Replicat as it applies transactions to the target system. Regardless of the Replicat that is being used, it is a best practice to enable the checkpoint table for the target system.

In order to do this, follow the below steps:

1. Log in to the Service Manager.
2. From the Service Manager main page, select the hyperlink for the port number associated with the Administration Service.
3. Open the Context menu, present on the top left corner of the Overview page and from the Context menu, select Configuration.
4. From the Database tab, select the database icon to log the target user to the database.
5. From the Checkpoint, select the plus (+) icon.
6. In the Checkpoint Table text box, provide the name of the checkpoint table. Checkpoint table entry must be two part or three part names, for making it more readable.
7. Click Submit.

 **Note**

For a multitenant database, you have to create a checkpoint table for each PDB.

Implementing Heartbeat Monitoring

The Automatic Heartbeat table is a key way to monitor latency within the Oracle GoldenGate framework. Heartbeat tables provide you a way to gauge the end-to-end throughput within the configuration and identify any potential bottlenecks in the network.

To implement the Automatic Heartbeat table, you should perform the following steps on both source and target database:

1. Log in to the ServiceManager.
2. From the SeviceManager main page, select the hyperlink for the port number associated with the Administration Service.
3. Open the context menu available on the top left corner of the Overview page.
4. From the context menu, select Configuration.
5. From the first tab - Database, select the database icon, to log in to the database.
6. From Heartbeat section, select the plus (+) icon.
7. Adjust the Frequency, Retention, and Purge Frequency for the heartbeat table.
8. Click Submit.

For more information on the Automatic Heartbeat functionality, refer to [Monitoring Oracle GoldenGate Processing](#) documentation.

Configuring Capture Support

Before you can begin replication, you have to set up the capture process. The capture process is also known as Extract. Oracle GoldenGate Microservices supports three type of Extracts. They are:

- Classic Extract
- Integrated Extract
- Initial Load Extract

To decide on which type of capture to use, refer to [Deciding Which Capture Method to Use](#) section of *Using Oracle GoldenGate for Oracle Database Guide*.

To build any of these Extracts, perform the following steps in Oracle GoldenGate Microservices:

1. Log in to the Service Manager console.
2. From the Service Manager main page, select the hyperlink for the port number associated with the Administration Service.
3. From the Overview page, under Extracts, select the plus (+) icon. Add Extract wizard appears.
4. In the Add Extract wizard, select the Extract Type and then click Next.
5. Provide the details for the Extract in the Extract Options and then Click Next.
6. In the Parameter File option, provide the details needed for Extract to run.
7. Click Create and Run.

 **Note**

The Classic Extract is still available for use but has been deprecated as of Oracle GoldenGate 18c release. Additionally, there is no capture support for Autonomous Database or Autonomous Transaction Processing platforms.

Configuring Apply Support

The apply process for replication, also known as Replicat, is very easy and simple to configure. There are five types of Replicats supported by the Oracle GoldenGate Microservices and these Replicats are:

- Integrated Replicat
- Non-Integrated Replicat
- Coordinated Replicat
- Parallel Integrated Replicat
- Parallel Non-Integrated Replicat

To decide on which Replicat to use, refer to [Deciding Which Apply Method to Use](#) section of *Using Oracle GoldenGate for Oracle Database Guide*.

To build any of these Replicats you can perform the following steps with Oracle GoldenGate Microservices:

1. Log in to the Administration Server.
2. From the Overview page, under Replicats, select the plus (+) icon and Add Replicat wizard is displayed.
3. In the Add Replicat wizard, select the Replicat Type and then click Next.
4. On the Replicat Options, provide the details for the Replicat and then click Next.
5. In the Parameter File, provide the details needed for the Replicat to run.
6. Click Create and Run.

Note

You can use only the Non-Integrated Replicat or Non-Integrated Parallel Replicat for replication to Autonomous Data Warehouse and Autonomous Transaction Processing.

Connecting to Data Resources

Learn about different methods of connecting Oracle GoldenGate data sources and targets. It includes the following connection types:

- [Connecting to Oracle Database \(on-premises\)](#)
- [Architectures](#)
- [Connecting to Oracle Database as a Service \(DBaaS\)](#)
- [Connecting to Oracle Autonomous Data Warehouse/Autonomous Transaction Processing](#)

Connecting to Oracle Database (on-premises)

You can use Oracle GoldenGate Microservices on Marketplace to remotely capture from and apply data to on-premises Oracle database resources. This allows you to enable replication and centrally manage the replication processes.

Use Cases for Replication

You can use Oracle GoldenGate Microservices to replicate data between data resources in the following use cases:

- Migrations
- Data Distribution
- Real-Time Data Warehousing
- Operational Reporting

Replicating Data from On-premises

Prerequisites

Ensure that the following are set up before you begin replication:

- Oracle GoldenGate Microservices
- Source Database
- Target Database

To move data from on-premises to the cloud or from on-premises to on-premises, perform the following tasks :

- [Configure Oracle Database for Replication](#)
- [Configure Oracle GoldenGate Microservices Compute Node](#)

Configure Oracle Database for Replication

To prepare your Database as a Service (DBaaS) instance for replication, perform the following tasks:

1. Configure Logging Properties

- Enable Supplemental Logging

2. Enable Oracle GoldenGate within the Oracle Database

- Update parameter for `enable_goldengate_replication`

For more details, refer to [Preparing the Database for Oracle GoldenGate](#) documentation.

Configure Oracle GoldenGate Microservices Compute Node

To connect the Microservices Compute Node to the on-premises databases, edit the `tnsnames.ora` file and point the entry to your database resources.

By default, the environment variable `TNS_ADMIN` is pre-configured for each deployment. But the files `tnsnames.ora` or `sqlnet.ora` are not readily available on the compute node. You need to create the files or copy them from an existing file. You can locate the files `tnsnames.ora` or `sqlnet.ora` in the pre-configured location as illustrated in the below table:

Oracle Database Version	Directory
Oracle 11g	/u02/deployments/<deployment>/etc
Oracle 12c	/u02/deployments/<deployment>/etc
Oracle 18c	/u02/deployments/<deployment>/etc
Oracle 19c	/u02/deployments/<deployment>/etc

Note

- If you are using Oracle Autonomous products, the `tnsnames.ora` file is included in your `Client_Credentials.zip` file. You have to edit this `tnsnames.ora` file.
- If you want your networking directory structure to be consistent with other Oracle products, you must append `/network/admin` to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

Architectures

Oracle GoldenGate on Oracle Cloud Marketplace enables you to work with many existing and new architectures.

This chapter helps you with a few examples and steps required to set up and configure the architectures. The supported architectures are:

- On-premises to Oracle Cloud (Marketplace)
- Oracle Cloud to Oracle Cloud (Marketplace to Marketplace)

Securely Connecting Oracle GoldenGate Microservices On Premise to Oracle GoldenGate Microservices on Marketplace

You can connect your on-premises Oracle GoldenGate Microservices architecture to the Oracle GoldenGate Microservices architecture on Oracle Cloud Marketplace using the following methods:

- Connecting through Public IP Address

- Connecting through IPSec VPN
- Connecting through FastConnect

The following section helps you to connect through public IP address. To use the IPSec VPN or the FastConnect approach, refer to the respective [VPN Connect](#) and [FastConnect](#) documentation.

Connecting Through Public IP Address

By default, we configure Oracle GoldenGate Microservices on Oracle Cloud Marketplace behind the Nginx Reverse Proxy. This simplifies the architecture on Oracle Cloud and also makes the deployment secure. When you connect over a public IP address, on-premises Oracle GoldenGate Microservices architecture has to be secure.

When the deployment is secure and in order to connect to it securely, perform the following steps on the on premise machine and the Oracle Cloud compute node:

In the On Premise Machine

1. Test the connection to Oracle Cloud Compute Node using OpenSSL.

```
$ openssl s_client -connect <public id address>:443
```

2. Update the on premise Oracle GoldenGate Microservices with the same SSH Key that is used to build the Oracle GoldenGate Microservices on Oracle Cloud Marketplace compute node.

Copy your private SSH key to the .ssh directory for the Oracle user on the Oracle GoldenGate Microservices environment.

```
$ su - oracle
$ cd .ssh
$ scp ....
$ sudo chmod 700 ~/.ssh/<private ssh key>
```

3. Copy the `ogg.pem` file from the Oracle GoldenGate Microservices compute node to the on premise environment.

- a. Copy the `ogg.pem` to the local machine.

```
$ scp opc@<public ip address>:/etc/nginx/ogg.pem
```

- b. Check if the `ogg.pem` file is present on the local machine through the following command:

```
$ ls -a
```

4. Update the local wallet for Oracle GoldenGate Microservices Distribution Service with the `ogg.pem` file. You have to assign it as a trusted certificate.

- a. Find the required wallet.

```
$ ps -ef | grep -i distsrvr
```

i. Use the listed dat file.

```
$ cat <dat file> | python -m json.tool
```

ii. Get the wallet location for the Distribution Service.

For example: \$DEPLOYMENT_HOME/etc/ssl/<distribution service wallet>

b. Update the wallet.

```
$ $OGG_HOME/bin/orapki wallet add -wallet <wallet directory> -  
trusted_cert -cert <certificate file path> -pwd <wallet password>
```

For example:

```
$OGG_HOME/bin/orapki wallet add -wallet /opt/app/oracle/  
gg_deployments/on-premises/etc/ssl/DistroClient -trusted_cert -cert /  
home/oracle/oci_cert.pem -pwd *****
```

c. Check the wallet.

```
$ $OGG_HOME/bin/orapki wallet display -wallet <wallet directory> -pwd  
<wallet password>
```

For example:

```
$OGG_HOME/bin/orapki wallet display -wallet /opt/app/oracle/  
gg_deployments/on-premises/etc/ssl/DistroClient -pwd *****
```

d. As root user, update the on premise /etc/hosts file. You can find the information that needs to go in to the /etc/hosts file on the Oracle GoldenGate Oracle Cloud Compute Node Instance in the Details page. The required information includes:

- Public IP Address
- Internal FQDN

From the Internal FQDN, you have to use the short hostname as well.

e. Stop the Microservices components. Any running Extracts and/or Replicats will not be affected.

i. Stop the deployment

- Login to ServiceManager.
- Under Deployments, click Action, to stop the deployment.

ii. Stop the ServiceManager.

- Login to ServiceManager.
- Under Deployments, click Action, to restart ServiceManager.

ⓘ Note

While restarting ServiceManager, navigate to the command prompt and issue one of the following commands:
Manually:

```
$ cd $DEPLOYMENT_HOME/bin  
$ ./stopSM.sh  
$ ./startSM.sh
```

Daemon, as root user:

```
service restart OracleGoldenGate.service
```

f. Start the Microservices Components.

i. Start the Deployment

- Login to ServiceManager.
- Under Deployments, click Action, to start the deployment.

On Oracle Cloud Compute Node

Follow the below steps from the Administration Service tab on the Oracle GoldenGate Microservices configuration page on Oracle Cloud:

1. Create a user, who can login to the environment and connect to the Receiver Service.
 - Login to the Administration Service. For this, you will need the oggadmin password available in the `ogg-credentials.json` file, unless you changed it after the initial setup.

```
https://<public id address>/<deployment name>/adminsrvr
```

For example-

```
https://<public ip address>/OCI-BASE/adminsrvr
```

2. Open the context menu and select the Administrator option.
3. Click the plus (+) icon, to add a new user with Operator role. This account is used by the on premise Distribution Service to login through the Reverse Proxy.

Provide the following information:

Username: streamnetwork
Role: Operator
Type: Basic
Info: Network User
Password: *****
Verify Password: *****

On Premise Machine

In the on premise machine:

- From the Administration Service tab on the on premise environment, create an alias that can connect to the network user created in the previous step. This alias is used by the Distribution Service to connect to the Receiver Service on Oracle Cloud.
 - Login to the Administration Service.
 - Open the context menu and select Configuration.
 - Click the plus (+) icon, to add a new credential.
 - Add a new **Credential** by providing the following information:

Credential Domain: Network
Credential Alias: streamnetwork
User ID: streamnetwork
Password: *****
Verify Password: *****

 **Note**

You cannot validate the new credential, as it is not logged in to the database.

- In the Distribution Service on the on-premise environment,

- Provide information needed for the Distribution Path. The basic information required are:

Path Name: OP2OCI
Reverse Proxy Enabled: Toggle to on
Use Basic Authentication: Toggle to on
Source: Select Extract and provide source trail file info
Target:

- Keep the WSS protocol
- Provide the Hostname of the OCI Compute node - IP will not work
- Provide remote trail file name
- Provide Deployment name
- Domain: Network
- Alias: securitynetwork
- Trail Size (MB): set to desired size

- Click **Create** or **Create and Run**.

Connecting to Oracle Database as a Service (DBaaS)

You can use Oracle GoldenGate Microservices on Marketplace to remotely capture and apply data to Oracle Cloud Infrastructure (OCI) Database as a Service (DBaaS) resources. This allows you to enable replication in a scalable fashion, centralize the point of management, and enable replication between cloud services.

Replicating Data to OCI Database as a Service Prerequisites

Ensure you have the following prerequisites before replicating data from on-premises:

- Oracle GoldenGate Microservices on Marketplace
- Source Database
- Oracle Cloud Infrastructure (OCI) Database as a Service (DBaaS) Instance

To move data from on-premises to the cloud or from on-premises-to-on-premises, perform the following tasks :

- [Configure Oracle Database for Replication](#)
- [Configure Oracle GoldenGate Microservices Compute Node](#)

Configure Oracle Database for Replication

To prepare your Database as a Service (DBaaS) instance for replication, perform the following tasks:

1. Configure Logging Properties
 - Enable Supplemental Logging
2. Enable Oracle GoldenGate within the Oracle Database
 - Update parameter for `enable_goldengate_replication`

For more details, refer to [Preparing the Database for Oracle GoldenGate](#) documentation.

Configure Oracle GoldenGate Microservices Compute Node

To establish connection from the Oracle GoldenGate Microservices Compute Node to your on-premises database, you must edit the `tnsnames.ora` file and point the entry to your database resources.

By default, the environment variable `TNS_ADMIN` is pre-configured for each deployment. But the files `tnsnames.ora` or `sqlnet.ora` are not readily available on the compute node. You need to create the files or copy them from an existing file. You can locate the files `tnsnames.ora` or `sqlnet.ora` in the pre-configured location as illustrated in the below table:

Table 2-1 Oracle Database Client directories

Oracle Database Version	Directory
Oracle 11g	/u02/deployments/<deployment>/etc
Oracle 12c	/u02/deployments/<deployment>/etc
Oracle 18c	/u02/deployments/<deployment>/etc
Oracle 19c	/u02/deployments/<deployment>/etc

Note

- If you are using Oracle Autonomous products, the `tnsnames.ora` file is included in your `Client_Credentials.zip` file. You can use this `tnsnames.ora` file or copy its contents to your `tnsnames.ora` file located in `$ORACLE_HOME/network/admin`.
- If you want your networking directory structure to be consistent with other Oracle products, you must append `/network/admin` to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

Connecting to Oracle Autonomous Data Warehouse/Autonomous Transaction Processing

You can replicate data to Oracle Autonomous Data Warehouse Cloud Service (ADWCS) or Autonomous Transaction Processing (ATP) by using Oracle GoldenGate Microservices on Oracle Cloud Marketplace. The steps described in this section, streamlines the approach for making a remote connection to Oracle Autonomous Database Warehouse Cloud Service (ADWCS).

For more information, refer to [Replicating Data to the Autonomous Database](#) section of *Using Oracle GoldenGate for Oracle Database* guide.

Use Cases for Replicating to Autonomous Database

Use Oracle GoldenGate Microservices to replicate data to the Autonomous Data Warehouse for:

- Real-time Data Warehousing
- Operational Reporting

Replicating Data to Autonomous Data Warehouse

Prerequisites:

The following prerequisites are essential for replicating data to Autonomous Data Warehouse:

- Oracle Autonomous Data Warehouse Cloud Service
- Your source database

To deliver data to the Autonomous Database using Oracle GoldenGate Microservices, perform the following tasks:

- [Configure the Autonomous Data Warehouse for Replication](#)
- [Autonomous Database Client Credentials](#)
- [Configure Oracle Goldengate Microservices for Replication](#)

Configure the Autonomous Data Warehouse for Replication

Unlock the Pre-created Oracle GoldenGate User (ggadmin)

Perform the following steps to configure the Autonomous Data Warehouse for Replication:

1. Unlock and change the password for the pre-created Oracle GoldenGate user (ggadmin) within the Autonomous Data Warehouse. Use any SQL client tool to unlock the account.

For more details, refer to [About Connecting to an Autonomous Data Warehouse Instance](#) section of *Using Oracle Autonomous Data Warehouse* guide.

```
SQL> select * from dba_users order by username;
SQL> alter user ggadmin identified by <password> account unlock;
```

2. Check whether the parameter `enable_goldengate_replicaton` is set to true. If not, then modify the parameter.

```
SQL> select * from v$parameter where name =
  'enable_goldengate_replication';
SQL> alter system set enable_goldengate_replication = true scope=both;
```

Create Target Schema

Complete the following steps to create schema and target objects that can be used in replication. This schema and associated objects does not support DDL replication.

1. Create a new application user/schema. This user/schema stores the target objects for replication.

 **Note**

`appadmin` is an example user.

```
SQL> create user appadmin identified by ****
SQL> grant create session, resource, create view, create table to appadmin;
SQL> alter user appadmin quota unlimited on data;
```

2. Connect to the Oracle Autonomous Data Warehouse Cloud database as user/schema and create your application tables.

Autonomous Database Client Credentials

Obtain the Autonomous Database Client Credentials

To establish connection to your Autonomous Database, you must download the client credential files from the Autonomous Database Service Console. For more information, see [Downloading Client Credentials](#) section of *Using Oracle Autonomous Data Warehouse* guide .

 **Note**

If you do not have administrator access to the Autonomous Database, ask your service administrator to download and provide the credential files to you. Once you have the credential files for your Autonomous Database, you should upload the zip file to the Oracle GoldenGate Compute Node.

Perform the following steps to obtain the Oracle Autonomous Data Warehouse Cloud account details:

1. Log in to your Oracle Autonomous Data Warehouse Cloud account.

2. From the `Instance` page, click the menu option for the Autonomous Database instance and select `Service Console`.
3. Log in to the `Service Console` using the `admin` username, and its associated password.
4. In the `Service Console`, click the `Administration` tab.
5. Click `Download Client Credentials`.
6. Enter a password to secure your credentials zip file and click `Download`.
7. Save the credentials zip file to your local system.

Move Client Credentials to Oracle GoldenGate Compute Node

In order to establish a connection from Oracle GoldenGate to the Autonomous Data Warehouse, you need to move the client credentials to Oracle GoldenGate Compute Node. The following steps will illustrate how to move the credential zip file from your machine to Oracle GoldenGate Compute Node.

1. Connect to the Oracle GoldenGate Classic Compute Node using SSH and `opc` user credentials.

```
ssh -i <private_key> opc@<public_ip_address>
```

2. Create a staging directory and grant the essential permissions and then exit the session.

```
$ mkdir stage  
$ exit
```

3. Copy the credentials zip file to the Oracle GoldenGate Classic Compute Node.

```
$ scp ./<credential_file>.zip opc@<public_id_address>:~/stage
```

4. Connect to the Oracle GoldenGate Classic Compute Node.

```
ssh -i <private_key> opc@<public_ip_address>
```

5. Verify whether the credentials zip file is available in the stage location.

```
$ cd ~/stage  
$ ls -ltr
```

Configure Oracle Goldengate Compute Node with Autonomous Client Credentials

After moving the ADWC Client Credentials to the Oracle GoldenGate Compute Node, you have to install the necessary files and ensure you have a connection to the Autonomous Data Warehouse. The following steps will help you configure the required SQL*Net components:

1. Log in to the Oracle GoldenGate Classic Compute Node using SSH and the `opc` user credentials.

```
ssh -i <private_key> opc@<public_ip_address>
```

2. Unzip the client credentials file into a temporary directory.

```
unzip ./<credential_file>.zip -d ./client_credentials
```

3. Copy the `sqlnet.ora` and `tnsnames.ora` files to the location of your `TNS_ADMIN`.

```
$ cd ~/stage/client_credentials  
$ cp ./sqlnet.ora /u02/deployments/<deployment>/etc  
$ cp ./tnsnames.ora /u02/deployments/<deployment>/etc
```

 **Note**

If you want your networking directory structure to be consistent with other Oracle products, you must append `/network/admin` to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

4. Edit the `sqlnet.ora` file and replace the directory parameter with the location of the information pointing to the location where the client credentials were unzipped.

```
$ cd /u02/deployments/<deployment>/etc  
$ vi ./sqlnet.ora
```

Change `?/network/admin` to `/home/opc/stage/client_credentials`.

5. For testing purposes, set the `TNS_ADMIN` and `ORACLE_HOME` environment variables at the operating system level.

 **Note**

The Oracle GoldenGate Deployment(s) use the `ORACLE_HOME` and `TNS_ADMIN` environment variables that are set per deployment.

```
$ export ORACLE_HOME=/u01/app/client/<oracle version>  
$ export TNS_ADMIN=/u02/deployments/<deployment>/etc
```

6. Test the connection to Autonomous Data Warehouse by connecting to one of the entries in the `tnsnames.ora` file.

```
$ cd $ORACLE_HOME/bin  
$ ./sqlplus appadmin/**********@orcladw_low
```

Configure Oracle Goldengate Microservices for Replication

Perform the following steps for establishing a successful connection to the Autonomous Data Warehouse with Oracle GoldenGate Microservices.

Add Oracle GoldenGate Credential to connect to Autonomous Data Warehouse

To add Oracle GoldenGate Credential details, to connect to Autonomous Data Warehouse:

1. Log in to the Service Manager using the password for `oggadmin`.
2. From the Service Manager main page, select the hyperlink for the port number associated with the Administration Service.

3. Open the context menu in the top left corner of the Overview page.
4. From the context menu, select Configuration.
5. From the Database tab, click the plus (+) icon, to add a new credential.
6. Provide the following information and click Submit.

Credential Domain: [Defaults to OracleGoldenGate]

Credential Alias: [Name of the Alias]

User ID: ggadmin@<adw_tnsnames_reference>

Password: [Password for ggadmin]

Verify Password: [Password for ggadmin]

7. Test the connection to the Autonomous Data Warehouse by clicking the Log in Database icon after the credential has been added.

Part III

Using Oracle GoldenGate Classic on Oracle Cloud Marketplace

The Oracle GoldenGate Classic Architecture provides the processes and files required to effectively move data across a variety of topologies. These processes and files form the main components of the Classic Architecture. You can find Oracle GoldenGate Classic Architecture under the same marketplace listing as the Microservices Architecture on Oracle Cloud Marketplace. You can use Oracle GoldenGate Classic as an optional architecture to Microservices architecture. Oracle GoldenGate Classic Architecture helps you to enable the heterogeneous nature of the Oracle GoldenGate product and align all the heterogeneous and Big Data platforms.

This part provides the high level steps used for building an Oracle Cloud Infrastructure Compute Node that supports Oracle GoldenGate Classic Architecture platforms. It contains the following chapters:

- [Oracle GoldenGate Classic for Oracle](#)
- [Oracle GoldenGate Classic for DB2 for z/OS](#)
- [Oracle GoldenGate Classic for Big Data](#)
- [Oracle GoldenGate Classic for SQL Server](#)
- [Oracle GoldenGate Classic for MySQL](#)

Oracle GoldenGate Classic for Oracle

Deploying Oracle GoldenGate Classic Architecture for Oracle Database

By using Oracle GoldenGate Classic Architecture on Marketplace, you can deploy Oracle GoldenGate in an off-box architecture, which means you can run and manage your Oracle GoldenGate deployment from a single location.

Here are the prerequisites required to deploy Oracle GoldenGate Classic:

- [Oracle Cloud Account](#)
- Access to compute node resources within Oracle Cloud

Finding Oracle GoldenGate within the Oracle Cloud Marketplace

To launch Oracle GoldenGate Classic, from Oracle Cloud Marketplace follow these steps:

1. Log in to [Oracle Cloud Marketplace](#).
2. From the Oracle Cloud Marketplace home page, use the search box under **Applications** and search for the keyword **GoldenGate**.
3. From the search results, select **Oracle GoldenGate**.

Launching Oracle GoldenGate Classic

The following steps help you to launch Oracle GoldenGate Classic on Oracle Cloud Marketplace using the TerraForm stack:

1. From the Application page, select **Get App**.
2. Select **OCI Region** or Log in using your OCI account login credentials.
 - OCI Region – Select the OCI Region for which you want to install the compute node and click **Create Stack**.
3. In the **Sign In to the Cloud Tenant** page, provide the OCI tenant details.
4. Sign in to the Identity provider.
5. On the **Oracle GoldenGate 19c for Oracle** page, provide the following information:
 - **Select Version** - It provides a list of versions that are available in 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** - This checkbox is selected by default. Oracle recommends to review the licenses before proceeding.
 - **Launch Stack** - It launches the stack in the OCI environment.

6. Fill in the **Stack** Information, which includes the name of the stack, description and tags. The details of the compartment where the compute node is built are pre-populated and are not editable. Click **Next**.
7. Fill in the required details to **Configure Variables**. This information is required to build the compute node for Oracle GoldenGate Classic.
 - Name for New Resources -
 - a. **Display Name** – Display name used to identify all new OCI resources.
 - b. **Host DNS Name** – Name of the Domain Name Service for the new compute node.
 - Network Settings -
 - a. **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.
 - b. **Network Compartment** (optional) – Compartment for new or existing network resources.
 - c. **VCN** (optional) – Existing VCN to use for the newly created instance if you are not creating a new network.
 - d. **Subnet** (optional) – Existing subnet to use for the newly created instance if you are not creating a new network.
 - Instance Settings -
 - a. **Compute Shape** – Shape of new compute instance. Supported shapes are VM.Standard2.4, VM.Standard2.8, VM.Standard2.16 and VM.Standard2.24.
 - b. **Assign Public IP** – Select this check box to indicate if the new VM should have a public IP address.
 - c. **Availability Domain** – The availability domain for the new Oracle GoldenGate instance.
 - d. **Custom Volume Sizes** – Select this check box to customize the size of new block storage volumes.
 - **SSH Public Key** - Public Key used for allowing SSH access as the `opc` user
8. Click **Next**.
9. On the **Review** page, review the information you provided and then click **Create**.
10. After clicking Create, you are navigated to the **Stacks Job Details** page. From this page, you can monitor the creation of compute node.
11. Once you see the build successful message from the Job Details page, you can now view the Oracle GoldenGate Classic compute node under **Compute -> Instances**.

Getting Started with Oracle GoldenGate Classic Compute Node

To access your new Oracle GoldenGate deployment, log in to the compute node and access Oracle GoldenGate Software Command Interface (GGSCI). For this, you need to identify the

public IP address of the compute node where Oracle GoldenGate Classic is running. The following steps help you to identify the public IP address:

1. Log in to your Oracle Cloud Console.
2. Select **Compute -> Instances**.
3. Select the hyperlink name of the compute node. The public IP Address is listed under **Primary VNIC Information**.
4. To access the compute node where Oracle GoldenGate is running, connect as the `opc` account using SSH. For more information on how to access a node using the `opc` account refer to [Connecting to an Instance](#).

A public SSH key is specified as a part of the deployment process and you can use the private key when you are connecting to the Oracle GoldenGate instance. The following example illustrates how you connect to the Oracle GoldenGate compute node:

```
ssh -i <private-key-filename> opc@<public-id-address>
```

Running GGSCI

Oracle GoldenGate Classic allows you to quickly access any of the Oracle GoldenGate environments without the need of setting up environment variables per installation.

A custom utility called `ggsci` is available in the `/usr/local/bin` directory, which helps you to ensure correct environment variables are created in the environment. But for using this, you have to set a mandatory parameter. For example, for Oracle Database 19c:

```
-bash-4.2$ ./ggsci oracle19
```

```
Oracle GoldenGate Command Interpreter for Oracle
Version 19.1.0.0.1 OGGCORE_19.1.0.0.0_PLATFORMS_190524.2201_FBO
Linux, x64, 64bit (optimized), Oracle 19c on May 25 2019 15:07:20
Operating system character set identified as UTF-8.
```

```
Copyright (C) 1995, 2019, Oracle and/or its affiliates. All rights reserved.
```

```
GGSCI (ogg19ccls) 1>
```

If you did not mention the Oracle Database version in the command, then it displays the usage information with the required details as follows:

```
-bash-4.2$ ./ggsci
Usage: ./ggsci <oracle-version>
Example: ./ggsci oracle11, oracle12, oracle18, oracle19
```

 **Note**

Do not run the GGSCI command from the `/u01/app/ogg` directory.

Creating Oracle GoldenGate Subdirectories

You can configure Oracle GoldenGate Classic after setting the environment variables. To create the subdirectories needed for Oracle GoldenGate, navigate to the Oracle GoldenGate

home and access Oracle GoldenGate Software Command Interface (GGSCI). Create the subdirectories as described in [Installing Oracle GoldenGate User Guide](#).

```
$ cd $OGG_HOME
$ ./ggsci
GGSCI> create subdirs
```

After creating the sub-directories, you can find them in `/home/opc/oracle<##>` directory. Although they appear in `/home/opc/oracle<##>` location, the contents are written to `/u02/deployments/<directory>`. This helps you to improve tolerance, if something had happened to the Oracle GoldenGate Compute Node.

Getting Started with Oracle GoldenGate Classic for Oracle Database

After deploying Oracle GoldenGate Classic on the Oracle Cloud Marketplace, you can access the latest release of Oracle GoldenGate Classic.

Before you can start using Oracle GoldenGate Classic, there are a few tasks that you must perform to ensure that your environment is complete and ready to replicate your data. Before you begin data replication, you must perform the following tasks:

Topics:

- [Configuring Source or Target Database for Replication](#)
- [Establishing Connectivity with Oracle GoldenGate Classic](#)
- [Updating Network Related Files for Oracle GoldenGate Classic](#)
- [Creating Database Credentials for Oracle GoldenGate Classic](#)
- [Adding SchemaTrandata for Oracle GoldenGate Classic](#)
- [Creating Checkpoint Table for Oracle GoldenGate Classic](#)
- [Enabling Heartbeat Table for Oracle GoldenGate Classic](#)
- [Configuring Capture Support for Oracle GoldenGate Classic](#)
- [Configuring Apply Support for Oracle GoldenGate Classic](#)

Configuring Source or Target Database for Replication

Before you can begin replicating, you should prepare the source or target database to support Oracle GoldenGate. To prepare your database, follow the steps listed in [Preparing the Database for Oracle GoldenGate](#) section of *Using Oracle GoldenGate for Oracle Database* guide. The steps listed in the *Using Oracle GoldenGate for Oracle Database* guide helps you to enable logging and kernel parameters, set the flashback query and manage server resources.

Establishing Connectivity with Oracle GoldenGate Classic

The Oracle GoldenGate Classic on Marketplace compute node acts as a hub where you can manage your connections to source and target database. To do this, establish a network connection between the compute node and your source and target database. The compute node is pre-configured with the Oracle Database Client software and is ready to use.

You have to provide `sqlnet.ora` and `tnsnames.ora` files in the `TNS_ADMIN` directory, to establish connection between source or target database and the compute node. On a deployment basis, the `TNS_ADMIN` directory has been established as `/u02/deployments/<deployment_name>`.

Updating Network Related Files for Oracle GoldenGate Classic

The Oracle GoldenGate on Marketplace Compute Node comes pre-configured with Oracle Client installed. In order to establish network configuration between the compute node and the source or target systems within your architecture, you have to add or update the network related files in the Oracle Client.

You can find these files in the `TNS_ADMIN` location and the recommended location is `/u02/deployment/oracle<##>/network/admin`. To create this structure per Oracle Client release, perform the following steps:

1. Connect to the Oracle GoldenGate Marketplace Compute Node, using `opc` user credentials.
2. Change directories to the location `/u02/deployment/oracle<##>`.
3. Create the directory structure for `TNS_ADMIN`

```
mkdir -p ./network/admin
```

Note

If you want the networking directory structure to be consistent with other Oracle products, you must append `/network/admin` to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

4. Copy the existing `sqlnet.ora` and `tnsnames.ora` files to the admin directory. You have to add or update these files on the compute node.

The following is an example of `sqlnet.ora` file:

```
NAMES.DIRECTORY_PATH= (TNSNAMES, ONAMES, HOSTNAME)
NAMES.DEFAULT_DOMAIN = ora.com
```

The following is an example of `tnsnames.ora` file:

```
TEST =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)(HOST = test2)(PORT = 1521))
    )
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = test)
  )
)
```

Creating Database Credentials for Oracle GoldenGate Classic

Use the credential store to store and use database credentials for the source and target databases for Oracle GoldenGate Classic. Use the `tnsnames.ora` file to connect to the required database.

To create database credentials, perform the following tasks:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Navigate to `/usr/local/bin` and execute `ggsci`.

```
$ cd /usr/local/bin  
$ ggsci oracle<##>
```

3. Add the credential store to the configuration.

```
ggsci> add credentialstore
```

4. Alter the credential store to add the required login credentials.

```
ggsci > alter credentialstore add user <user_id>@<tnsnames_string>  
password ***** alias <alias_name> domain OracleGoldenGate
```

5. Log in to the database to test the connection.

```
ggsci > dblogin useridalias <alias_name> domain OracleGoldenGate
```

Adding SchemaTrandata for Oracle GoldenGate Classic

After adding the credential for connecting to the source database, you must enable supplemental logging on the source schema. The following steps are used to add SchemaTrandata to the source schema:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Execute `GGSCI`.

```
$ cd /usr/local/bin  
$ ./ggsci oracle<##>
```

3. Connect to the source database using the credential store alias.

```
ggsci > dblogin useridalias <alias_name> domain OracleGoldenGate
```

4. Add supplemental logging to the schema using `schematradata`.

```
ggsci > add schematradata <schema>
```

① Note

If your source database is multitenant and you are connecting to the root using a c## user, from Oracle Database 12.1 and later, you have to specify the PDB database with the schema. i.e. < pdb > . < schema >.

Creating Checkpoint Table for Oracle GoldenGate Classic

Checkpoint tables contain the data necessary for tracking the progress of the Replicat as it applies transactions to the target system. Regardless of the Replicat that is being used, it is a best practice to enable the checkpoint table for the target system.

① Note

You can predetermine a default checkpoint table name by ensuring the parameter `CHECKPOINTTABLE <table name>` in the `GLOBALS` file.

To create a checkpoint table, follow these steps:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Execute `GGSCI`.

```
$ cd /usr/local/bin  
$ ./ggsci oracle<##>
```

3. Connect to the target database using the credential store alias.

```
ggsci > dblogin useridalias <alias_name> domain OracleGoldenGate
```

4. Add checkpoint table.

```
ggsci > add checkpointtable <schema>.checkpointtable
```

Enabling Heartbeat Table for Oracle GoldenGate Classic

The Automatic Heartbeat table is a key way to monitor latency within the Oracle GoldenGate framework. Heartbeat tables provide you a way to gauge the end-to-end throughput within the configuration and identify any potential bottlenecks in the network. For a multitenant database, you have to create a heartbeat table in each PDB to which OGG can apply data.

To implement the Automatic Heartbeat table, perform the following steps:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Execute GGSCI.

```
$ cd /usr/local/bin  
$ ./ggsci oracle<##>
```

3. Connect to the source or target database using the credential store alias.

```
ggsci > dblogin useridelalias <alias_name> domain OracleGoldenGate
```

4. Add automatic heartbeattable.

```
ggsci > add heartbeattable
```

For more information on the Automatic Heartbeat functionality, refer to [Monitoring Oracle GoldenGate Processing](#) documentation.

Configuring Capture Support for Oracle GoldenGate Classic

Before you can begin replication, you have to set up the capture process. The capture process is also known as Extract. Oracle GoldenGate Classic supports three type of Extracts. They are:

- Classic Extract
- Integrated Extract
- Initial Load Extract

To decide on which type of capture to use, refer to [Deciding Which Capture Method to Use](#) section of *Using Oracle GoldenGate for Oracle Database Guide*.

To build any of these Extracts, perform the following steps in Oracle GoldenGate Classic:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Execute GGSCI.

```
$ cd /usr/local/bin  
$ ./ggsci oracle<##>
```

3. Connect to the source database using the credential store alias.

```
ggsci > dblogin useridelalias <alias_name> domain OracleGoldenGate
```

4. Add an Extract.

- For Classic Extract -

```
ggsci > add extract <extract_name>, tranlog, begin now  
ggsci > add exttrail ./dirdat/aa, extract <extract_name>
```

- For Integrated Extract -

```
ggsci > add extract <extract_name>, integrated tranlog, begin now  
ggsci > register extract <extract_name>, database container (< pdb_name >)  
ggsci > add exttrail ./dirdat/aa, extract <extract_name>
```

- For Initial Load Extract -

```
ggsci > add extract <extract_name>, sourceistable
```

Configuring Apply Support for Oracle GoldenGate Classic

The apply process for replication, also known as Replicat, is very easy and simple to configure. There are five types of Replicats supported by Oracle GoldenGate Classic and these Replicats are:

- Integrated Replicat
- Classic Replicat
- Coordinated Replicat
- Parallel Replicat in Integrated Mode
- Parallel Replicat in Non-Integrated Mode

To decide on which Replicat to use, refer to [Deciding Which Apply Method to Use](#) section of *Using Oracle GoldenGate for Oracle Database Guide*.

To build any of these Replicats, perform the following steps on Oracle GoldenGate Classic:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Execute GGSCI.

```
$ cd /usr/local/bin  
$ ./ggsci oracle<##>
```

3. Connect to the source database using the credential store alias.

```
ggsci > dblogin useridalias <alias_name> domain OracleGoldenGate
```

4. Add a Replicat.

- For Classic Replicat:

```
ggsci > add replicat <replicat_name>, exttrail ./dirdat/ab,  
checkpointtable <schema>.<checkpointtable>
```

- For Coordinated Replicat:

```
ggsci > add replicat <replicat_name>, coordinated [maxthreads  
<number>], exttrail ./dirdat/ab, checkpointtable  
<schema>.<checkpointtable>
```

- For Parallel Replicat in Non-Integrated Mode:

```
ggsci > add replicat <replicat_name>, parallel, exttrail ./dirdat/ab,  
checkpointtable <schema>.<checkpointtable>
```

ⓘ Note

You can use only Classic Replicat or Parallel Replicat in Non-Integrated mode for replication to Autonomous Data Warehouse and Autonomous Transaction Processing.

Connecting to Data Resources for Oracle GoldenGate Classic

Learn about different methods of connecting Oracle GoldenGate data sources and targets. It includes the following connection types:

- [Connecting to Oracle Database \(on-premises\) for Oracle GoldenGate Classic](#)
- [Connecting to Oracle Database as a Service for Oracle GoldenGate Classic](#)
- [Connecting to Oracle Autonomous Data Warehouse/Autonomous Transaction Processing](#)

Connecting to Oracle Database (on-premises) for Oracle GoldenGate Classic

You can use Oracle GoldenGate Classic on Marketplace to remotely capture from and apply data to on-premises Oracle database resources. This allows you to enable replication and centrally manage the replication processes.

Use Cases for Replication

You can use Oracle GoldenGate Classic to replicate data between data resources in the following use cases:

- Migrations
- Data Distribution
- Real-Time Data Warehousing
- Operational Reporting

Replicating Data from On-premises

Prerequisites

Ensure that the following are set up before you begin replication:

- Oracle GoldenGate Classic
- Source Database
- Target Database

To move data from on-premises to the cloud or from on-premises to on-premises, perform the following tasks :

- [Configure Oracle Database for Replication](#)
- [Configure Oracle GoldenGate Classic Compute Node](#)

Configure Oracle Database for Replication

To prepare your on-premise Oracle database(s) for replication, perform the following tasks:

1. Configure Logging Properties
 - Enable Supplemental Logging

2. Enable Oracle GoldenGate within the Oracle Database
 - Update parameter for `enable_goldengate_replication`

For more details, refer to [Preparing the Database for Oracle GoldenGate](#) documentation.

Configure Oracle GoldenGate Classic Compute Node

To establish connection from the Oracle GoldenGate Classic Compute Node to your on-premises database, you must edit the `tnsnames.ora` file and point the entry to your database resources.

By default, the environment variable `TNS_ADMIN` is pre-configured for each deployment. But the files `tnsnames.ora` or `sqlnet.ora` are not readily available on the compute node. You need to create the files or copy them from an existing file. You can locate the files `tnsnames.ora` or `sqlnet.ora` in the pre-configured location as illustrated in the below table:

Table 3-1 Oracle Database Client directories

Oracle Database Version	Directory
Oracle 11g	<code>/u02/deployments/oracle<##>/network/admin</code>
Oracle 12c	<code>/u02/deployments/oracle<##>/network/admin</code>
Oracle 18c	<code>/u02/deployments/oracle<##>/network/admin</code>
Oracle 19c	<code>/u02/deployments/oracle<##>/network/admin</code>

 **Note**

- If you are using Oracle Autonomous product client credentials, then you have to edit the `tnsnames.ora` file.
- If you want your networking directory structure to be consistent with other Oracle products, you must append `/network/admin` to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

Connecting to Oracle Database as a Service for Oracle GoldenGate Classic

You can use Oracle GoldenGate Classic on Marketplace to remotely capture and apply data to Oracle Cloud Infrastructure (OCI) Database as a Service (DBaaS) resources. This allows you to enable replication in a scalable fashion, centralize the point of management, and enable replication between cloud services.

Replicating Data to OCI Database as a Service Prerequisites

Ensure you have the following prerequisites before replicating data from on-premises:

- Oracle GoldenGate Classic on Marketplace
- Source Database

- Oracle Cloud Infrastructure (OCI) Database as a Service (DBaaS) Instance

To move data from on-premises to the cloud or from on-premises-to-on-premises, perform the following tasks :

- [Configure Oracle Database for Replication](#)
- [Configure Oracle GoldenGate Classic Compute Node](#)

Configure Oracle Database for Replication

To prepare your on-premise Oracle database(s) for replication, perform the following tasks:

1. Configure Logging Properties
 - Enable Supplemental Logging
2. Enable Oracle GoldenGate within the Oracle Database
 - Update parameter for `enable_goldengate_replication`

For more details, refer to [Preparing the Database for Oracle GoldenGate](#) documentation.

Configure Oracle GoldenGate Classic Compute Node

To establish connection from the Oracle GoldenGate Classic Compute Node to your on-premises database, you must edit the `tnsnames.ora` file and point the entry to your database resources.

By default, the environment variable `TNS_ADMIN` is pre-configured for each deployment. But the files `tnsnames.ora` or `sqlnet.ora` are not readily available on the compute node. You need to create the files or copy them from an existing file. You can locate the files `tnsnames.ora` or `sqlnet.ora` in the pre-configured location as illustrated in the below table:

Table 3-2 Oracle Database Client directories

Oracle Database Version	Directory
Oracle 11g	/u02/deployments/oracle<##>/network/admin
Oracle 12c	/u02/deployments/oracle<##>/network/admin
Oracle 18c	/u02/deployments/oracle<##>/network/admin
Oracle 19c	/u02/deployments/oracle<##>/network/admin

Note

- If you are using Oracle Autonomous product client credentials, then you have to edit the `tnsnames.ora` file.
- If you want your networking directory structure to be consistent with other Oracle products, you must append `/network/admin` to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

Connecting to Oracle Autonomous Data Warehouse/Autonomous Transaction Processing

You can replicate data to Oracle Autonomous Data Warehouse Cloud Service (ADWCS) or Autonomous Transaction Processing (ATP) by using Oracle GoldenGate Classic on Oracle Cloud Marketplace. The steps described in this section, streamlines the approach for making a remote connection to Oracle Autonomous Database Warehouse Cloud Service (ADWCS).

For more information, refer to [Replicating Data to the Autonomous Database](#) section of *Using Oracle GoldenGate for Oracle Database* guide.

Use Cases for Replicating to Autonomous Database

Use Oracle GoldenGate Classic to replicate data to the Autonomous Data Warehouse for:

- Real-time Data Warehousing
- Operational Reporting

Replicating Data to Autonomous Data Warehouse

Prerequisites:

Ensure to have the following prerequisites before replicating data to Autonomous Data Warehouse:

- Oracle GoldenGate Classic
- Oracle Autonomous Data Warehouse Cloud Service
- Your source database

To deliver data to the Autonomous Database using Oracle GoldenGate Classic, perform the following tasks:

- [Configure the Autonomous Data Warehouse for Replication](#)
- [Autonomous Database Client Credentials](#)
- [Configure Oracle Goldengate Classic for Replication](#)

Configure the Autonomous Data Warehouse for Replication

Unlock the Pre-created Oracle GoldenGate User (ggadmin)

Perform the following steps to configure the Autonomous Data Warehouse for Replication:

1. Unlock and change the password for the pre-created Oracle GoldenGate user (ggadmin) within the Autonomous Data Warehouse. Use any SQL client tool to unlock the account. For more details, refer to [About Connecting to an Autonomous Data Warehouse Instance](#) section of *Using Oracle Autonomous Data Warehouse* guide.

```
SQL> select * from dba_users order by username;
SQL> alter user ggadmin identified by <password> account unlock;
```

2. Check whether the parameter `enable_goldengate_replicaton` is set to true. If not, then modify the parameter.

```
SQL> select * from v$parameter where name =
  'enable_goldengate_replication';
SQL> alter system set enable_goldengate_replication = true scope=both;
```

Create Target Schema

Complete the following steps to create schema and target objects that can be used in replication. This schema and associated objects does not support DDL replication.

1. Create a new application user/schema. This user/schema stores the target objects for replication.

 **Note**

appadmin is an example user.

```
SQL> create user appadmin identified by *****  
SQL> grant create session, resource, create view, create table to appadmin;  
SQL> alter user appadmin quota unlimited on data;
```

2. Connect to the Oracle Autonomous Data Warehouse Cloud database as user/schema and create your application tables.

Autonomous Database Client Credentials

Obtain the Autonomous Database Client Credentials

To establish connection to your Autonomous Database, you must download the client credential files from the Autonomous Database Service Console. For more information, see [Downloading Client Credentials](#) section of *Using Oracle Autonomous Data Warehouse* guide .

 **Note**

If you do not have administrator access to the Autonomous Database, ask your service administrator to download and provide the credential files to you. Once you have the credential files for your Autonomous Database, you should upload the zip file to the Oracle GoldenGate Compute Node.

Perform the following steps to obtain the Oracle Autonomous Data Warehouse Cloud account details:

1. Log in to your Oracle Autonomous Data Warehouse Cloud account.
2. From the **Instance** page, click the menu option for the Autonomous Database instance and select **Service Console**.
3. Log in to the Service Console using the admin username, and its associated password.
4. In the Service Console, click the **Administration** tab.
5. Click **Download Client Credentials**.
6. Enter a password to secure your credentials zip file and click **Download**.
7. Save the credentials zip file to your local system.

Move Client Credentials to Oracle GoldenGate Compute Node

In order to establish a connection from Oracle GoldenGate to the Autonomous Data Warehouse, you need to move the client credentials to Oracle GoldenGate Compute Node. The following steps will illustrate how to move the credential zip file from your machine to Oracle GoldenGate Compute Node.

1. Connect to the Oracle GoldenGate Classic Compute Node using SSH and `opc` user credentials.

```
ssh -i <private_key> opc@<public_ip_address>
```

2. Create a staging directory and grant the essential permissions and then exit the session.

```
$ mkdir stage  
$ exit
```

3. Copy the credentials zip file to the Oracle GoldenGate Classic Compute Node.

```
$ scp ./<credential_file>.zip opc@<public_id_address>:~/stage
```

4. Connect to the Oracle GoldenGate Classic Compute Node.

```
ssh -i <private_key> opc@<public_ip_address>
```

5. Verify whether the credentials zip file is available in the stage location.

```
$ cd ~/stage  
$ ls -ltr
```

Configure Oracle Goldengate Compute Node with Autonomous Client Credentials

After moving the ADWC Client Credentials to the Oracle GoldenGate Compute Node, you have to install the necessary files and ensure you have a connection to the Autonomous Data Warehouse. The following steps will help you configure the required SQL*Net components:

1. Log in to the Oracle GoldenGate Classic Compute Node using SSH and the `opc` user credentials.

```
ssh -i <private_key> opc@<public_ip_address>
```

2. Unzip the client credentials file into a temporary directory.

```
unzip ./<credential_file>.zip -d ./client_credentials
```

3. Copy the `sqlnet.ora` and `tnsnames.ora` files to the location of your `TNS_ADMIN`.

```
$ cd ~/stage/client_credentials  
$ cp ./sqlnet.ora /u02/deployments/<deployment>/etc  
$ cp ./tnsnames.ora /u02/deployments/<deployment>/etc
```

① Note

If you want your networking directory structure to be consistent with other Oracle products, you must append /network/admin to the directory structure. For this you have to change the environment variable `TNS_ADMIN`. For more information on Local Naming Parameters, refer to [Database Net Services Reference](#).

4. Edit the `sqlnet.ora` file and replace the directory parameter with the location of the information pointing to the location where the client credentials were unzipped.

```
$ cd /u02/deployments/<deployment>/etc  
$ vi ./sqlnet.ora
```

Change `?/network/admin` to `/home/opc/stage/client_credentials`.

5. For testing purposes, set the `TNS_ADMIN` and `ORACLE_HOME` environment variables at the operating system level.

① Note

The Oracle GoldenGate Deployment(s) use the `ORACLE_HOME` and `TNS_ADMIN` environment variables that are set per deployment.

```
$ export ORACLE_HOME=/u01/app/client/<oracle version>  
$ export TNS_ADMIN=/u02/deployments/<deployment>/etc
```

6. Test the connection to Autonomous Data Warehouse by connecting to one of the entries in the `tnsnames.ora` file.

```
$ cd $ORACLE_HOME/bin  
$ ./sqlplus appadmin/**********@orcladw_low
```

Configure Oracle Goldengate Classic for Replication

Perform the following steps for establishing a successful connection to the Autonomous Data Warehouse with Oracle GoldenGate Classic.

Add Oracle GoldenGate Credential to connect to Autonomous Data Warehouse

To add Oracle GoldenGate Credential details, to connect to Autonomous Data Warehouse:

1. Log into the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Navigate to `/usr/local/bin` and execute `ggsci`.

```
$ cd /usr/local/bin  
$ ggsci oracle<##>
```

3. Add the credential store to the configuration.

```
ggsci> add credentialstore
```

4. Alter the credential store to add the required login credentials.

```
ggsci > alter credentialstore add user <user_id>@<tnsnames_string>
password ***** alias <alias_name> domain OracleGoldenGate
```

5. Log in to the autonomous database to test the connection.

```
ggsci > dblogin useridalias <alias_name> domain OracleGoldenGate
```

Oracle GoldenGate Classic for DB2 for z/OS

Deploying Oracle GoldenGate Classic Architecture for DB2 for z/OS Platforms

Using Oracle GoldenGate Classic Architecture on Marketplace, you can deploy and manage your deployment from a single location.

Here are the prerequisites which are required to deploy Oracle GoldenGate Classic:

- [Oracle Cloud Account](#)
- Access to compute node resources within Oracle Cloud Tenant

Finding Oracle GoldenGate for Non-Oracle (DB2 for z/OS) within the Oracle Cloud Marketplace

The following steps help you to deploy Oracle GoldenGate for non-Oracle (DB2 for z/OS) from Oracle Cloud Marketplace:

1. Log in to [Oracle Cloud Marketplace](#).
2. From the Oracle Cloud Marketplace home page, use the search box under **Applications** and search for the keyword **GoldenGate**.
3. From the Search Results, select **Oracle GoldenGate for Non-Oracle**.

Launching Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS)

The following steps help you to launch Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) on Oracle Cloud Marketplace:

1. From the application page, select **Get App**.
2. Select **OCI Region** or Log in using your OCI account login credentials.
 - a. OCI Region – Select the OCI Region for which you want to install the compute node.
 - b. Click **Create Stack**.
3. In the **Sign In to the Cloud Tenant** page, provide the OCI tenant details.
You may be asked to sign in to your identity provider.
4. On the **Oracle GoldenGate 19c for Non-Oracle (DB2 for z/OS)** page, find the selection box and provide the following information:
 - a. **Select Version** – It provides a list of versions that are available in the listing. It is set to Oracle GoldenGate for Non-Oracle (BigData) by default.

- b. **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
 - c. **Terms of Use** – This check box is selected by default. Oracle recommends to review the licenses before proceeding with the instance creation.
 - d. **Launch Stack** – It launches the stack in the OCI environment.
5. Fill in the required **Stack Information**, which includes name of the stack, description and tags. The details of the compartment in which you have to create the instance is pre-populated and it is not editable. After filling in the required details, click **Next**.
6. Fill in the required details to **Configure Variables**. This information is required to build the compute node for Oracle GoldenGate for Non-Oracle (DB2 for z/OS):
 - a. Name for New Resources -
 - **Display Name** - Display Name used to identify all new OCI resources.
 - **Host DNS Name** – Domain Name Service for the new compute node.
 - b. Network Settings -
 - **Create New Network** – Select this check box, if you wish to create a new network resource.
 - i. If you select this check box, the **Create New Network** wizard appears allowing you to add and edit the new network information such as:
 - Network Compartment
 - New VCN DNS Name
 - New VCN CIDR
 - New Subnet DNS Name
 - New Subnet CIDR
 - ii. 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.
 - **Network Compartment** (optional) – Compartment for new or existing network resources.
 - **VCN** (optional) – Existing VCN to use for the newly created instance if you are not creating a new network.
 - **Subnet** (optional) – Existing subnet to use for the newly created instance if you are not creating a new network. The subnet that you have selected must match the same Availability Domain set in the Instance Settings.
 - c. Instance Settings -
 - **Availability Domain** – It specifies the availability domain for the newly created Oracle GoldenGate Instance. It must match the Subnet that you have selected in the Use Existing Network settings.
 - **Compute Shape** – Shape of new compute instance. Supported shapes are VM.Standard2.4, VM.Standard2.8, VM.Standard2.16, VM.Standard2.24
 - **Assign Public IP** – Indicates if the new VM should have a public IP address
 - **Custom Volume Sizes** – Use this checkbox to customize the size of new block storage volumes
 - d. **SSH Public Key** - Public Key for allowing SSH access as the 'opc' user.

7. Click **Next**.
8. On the Review page, review the information you provided then click **Create**.
9. After clicking **Create**, you will be taken to the **Stacks** page. You can monitor the creation of the compute node using this page.
10. Upon completion, you can view the Oracle GoldenGate for Non-Oracle (DB2 for z/OS) compute node under **Compute -> Instances**.

Getting Started with Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) Compute Node

After deploying Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS), log in to the compute node and access Oracle GoldenGate Software Command Interface (GGSCI). For this, you need to identify the public IP address of the compute node where Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) is running. The following steps help you to identify the public IP address:

- Log in to your Oracle Cloud Console.
- Navigate to **Compute -> Instances**.
- Select the hyperlink name of the compute node. The public IP address is listed under **Primary VNIC Information**.
- To access the compute node where Oracle GoldenGate is running, connect as the `opc` account using SSH. For more information on how to access a node using the `opc` account refer to [Connecting to an Instance](#).

A public SSH key is specified as a part of the deployment process and you can use the private key when you are connecting to the Oracle GoldenGate instance. The following example illustrates how you connect to the Oracle GoldenGate compute node:

```
ssh -i <private-key-filename> opc@<public-id-address>
```

Installing DB2 drivers

The required DB2 connection drivers are not pre-installed in Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) compute node. You have to install one of the following DB2 connection drivers before using Oracle GoldenGate (listed from lightest to heaviest) :

- IBM Data Server Driver for ODBC and CLI v10.5 or later
- IBM Data Server Runtime Client v10.5 or later
- IBM Data Server Client v10.5 or later
- DB2 Connect v10.5 or later

For more information on these drivers, refer to [Oracle GoldenGate Installation Guide](#)

Running GGSCI

Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) allows you to quickly access the GoldenGate Service Command Interface (GGSCI). After logging in to the compute node, you can find GGSCI from `home -> db2zos` directory (`/home/opc/db2zos`). From here, after setting the above DB2 Driver environment, you can execute GGSCI.

1. To start GGSCI, execute the following commands:

```
$ cd db2zos  
$ ./ggsci
```

2. In GGSCI, create sub directories:

```
ggsci>create subsirs
```

3. Start manager:

```
ggsci>start manager
```

Working with Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS)

After deploying Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) on Oracle Cloud Marketplace, you can access the latest release of Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS).

Before you can start using Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS), there are a few tasks that you must perform to ensure that your environment is complete and ready to replicate your data. Before you begin data replication, you must perform the following tasks:

Topics:

- [Configuring Source and Target Database for Replication](#)
- [Establishing Connectivity](#)
- [Configuring Capture Support](#)
- [Configuring Apply Support](#)

Configuring Source and Target Database for Replication

Before you can begin replicating data, you should prepare the source or target database to support Oracle GoldenGate. To prepare your database, follow the steps listed in [Installing Oracle GoldenGate for DB2 z/OS Databases](#) chapter of *Installing Oracle GoldenGate Guide*. The steps listed here help you to enable the requirements to capture and/or apply data to DB z/OS.

Establishing Connectivity

The Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) on Oracle Cloud Marketplace compute node acts as a hub where you can manage your connections to source and/or target DB2 databases. To do this, establish a network connection between OCI compute node and your source and/or target DB2 databases. For this, you have to install the required IBM DB2 connection libraries on the OCI compute node.

After installing the necessary IBM DB2 connection libraries, to establish connectivity from an Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) compute node to source and/or target DB2 databases, use the following options:

- IPSec VPN

- FastConnect

For establishing an IPSec VPN with OCI, refer to [IPSec VPN](#) documentation. For establishing a FastConnect connection, refer to the [FastConnect](#) documentation.

Configuring Capture Support

Before you can begin replication, you have to setup the capture process. The capture process is also known as Extract. For DB2 on z/OS platforms, Oracle GoldenGate Classic supports only Remote Extract.

Before installing Remote Extract, refer to [Installing Extract Components](#) section in *Installing Oracle Goldengate Guide*.

To build the remote extract, perform the following steps within Oracle GoldenGate for Non-Oracle:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Start GGSCI.

```
$ cd db2zos
$ ./ggsci
```

3. Connect to the source database using the credential store.

```
ggsci > alter credentialstore add user <user_id>@<connect_string> password
***** alias <alias_name> domain OracleGoldenGate
```

4. Add an Extract.

- Remote Extract

```
ggsci > add extract <extract_name>, tranlog, begin now
ggsci > add exttrail ./dirdat/aa, extract <extract_name>, megabytes 250
```

Configuring Apply Support

The apply process for replication, also known as Replicat, is very easy and simple to configure. To build a Replicat, perform the following steps with Oracle GoldenGate for Non-Oracle.

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Start GGSCI.

```
$ cd db2zos
$ ./ggsci
```

3. Add a Replicat.

```
ggsci > add replicat <replicat_name>, exttrail ./dirdat/ab,
checkpointtable <schema>.<checkpointtable>
```

Connecting to Data Resources

Connecting to DB2 z/OS (On-Premises)

You can use Oracle GoldenGate Classic for Non-Oracle (DB2 z/OS) on Oracle Cloud Marketplace to remotely capture and apply data to DB2 z/OS Database on-premises or across Oracle Cloud platforms. This allows you to enable replication between cloud services in a scalable fashion and centrally manage the replication processes.

Use Cases for Replication

Use Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) to replicate data between data resources in the following use cases:

- Migrations
- Data Distribution
- Real-Time Data Warehousing
- Operational Reporting

Prerequisites

Ensure that the following are set up before you begin replication:

- Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS)
- Oracle DB2 Source or Target Database

To move data from on-premises to the cloud or from on-premises to on-premises, perform the tasks listed in the following sections:

- [Configuring Oracle DB2 z/OS for Replication](#)
- [Configuring Oracle GoldenGate Classic for Non-Oracle \(DB2 z/OS\) Compute Node](#)

Configuring Oracle DB2 z/OS for Replication

To prepare your Oracle DB2 z/OS database for replication, perform the tasks that are outlined in [Installing Oracle GoldenGate for DB2 z/OS Databases](#) chapter of *Installing Oracle GoldenGate Guide* and [Preparing the DB2 for z/OS Databases for Oracle GoldenGate](#) chapter of *Using Oracle Goldengate for Heterogenous Databases Guide*.

Configuring Oracle GoldenGate Classic for Non-Oracle (DB2 z/OS) Compute Node

To establish connection from Oracle GoldenGate Classic for Non-Oracle (DB2 for z/OS) compute node to your on-premise DB2 z/OS database, you must establish connectivity between OCI and on-premise database. To do this, you can use the following options:

- IPSec VPN
- FastConnect

For establishing an IPSec VPN with OCI, refer to [IPSec VPN](#) documentation. For establishing a FastConnect connection, refer to the [FastConnect](#) documentation.

Oracle GoldenGate Classic for Big Data

Deploying Oracle GoldenGate Classic Architecture for Big Data and Cloud Platforms

Using Oracle GoldenGate Classic Architecture on Marketplace, you can deploy and manage your deployment from a single location.

Here are the prerequisites which are required before deployment:

- [Oracle Cloud Account](#)
- Access to compute node resources within Oracle Cloud Tenant

Finding Oracle GoldenGate for Big Data within the Oracle Cloud Marketplace

The following steps help you to deploy Oracle GoldenGate for Big Data from Oracle Cloud Marketplace:

1. Log in to [Oracle Cloud Marketplace](#).
2. From the Oracle Cloud Marketplace home page, use the search box under **Applications** and search for the keyword **GoldenGate**.
3. From the Search Results, select **Oracle GoldenGate for Big Data**.

Launching Oracle GoldenGate Classic for Big Data

The following steps help you to launch Oracle GoldenGate Classic for Big Data on Oracle Cloud Marketplace:

1. From the application page, select **Get App**.
2. Select **OCI Region** or Log in using your OCI account login credentials.
 - a. OCI Region – Select the OCI Region for which you want to install the compute node.
 - b. Click **Create Stack**.
3. In the **Sign In to the Cloud Tenant** page, provide the OCI tenant details.
You may be asked to sign in to your identity provider.
4. On the **Oracle GoldenGate for Big Data** page, find the selection box and provide the following information:
 - a. **Select Version** – It provides a list of versions that are available in the listing. It is set to Oracle GoldenGate for Big Data by default.
 - b. **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.

- c. **Terms of Use** – This check box is selected by default. Oracle recommends to review the licenses before proceeding with the instance creation.
 - d. **Launch Stack** – It launches the stack in the OCI environment.
5. Fill in the required **Stack Information**, which includes name of the stack, description and tags. The details of the compartment in which you have to create the instance is pre-populated and it is not editable. After filling in the required details, click **Next**.
6. Fill in the required details to **Configure Variables**. This information is required to build the compute node for Oracle GoldenGate for Big Data:
 - a. Name for New Resources -
 - **Display Name** - Display Name used to identify all new OCI resources.
 - **Host DNS Name** – Domain Name Service for the new compute node.
 - b. Network Settings -
 - **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 such as:
 - Network Compartment
 - New VCN DNS Name
 - New VCN CIDR
 - New Subnet DNS Name
 - New Subnet CIDR
 - c. Instance Settings -
 - **Availability Domain** – It specifies the availability domain for the newly created Oracle GoldenGate Instance. It must match the Subnet that you have selected in the Use Existing Network settings.
 - **Compute Shape** – Shape of new compute instance. Supported shapes are VM.Standard2.4, VM.Standard2.8, VM.Standard2.16, VM.Standard2.24
 - **Assign Public IP** – Indicates if the new VM should have a public IP address.
 - **Custom Volume Sizes** – Use this checkbox to customize the size of new block storage volumes.
 - d. **SSH Public Key** - Public Key for allowing SSH access as the 'opc' user.
7. Click **Next**.
8. On the Review page, review the information you provided then click **Create**.

9. After clicking **Create**, you will be taken to the **Stacks** page. You can monitor the creation of the compute node using this page.
10. Upon completion, you can view the Oracle GoldenGate for Big Data compute node under **Compute -> Instances**.

Getting Started with Oracle GoldenGate Classic for Big Data Compute Node

After deploying Oracle GoldenGate Classic for Big Data, log in to the compute node and access Oracle GoldenGate Software Command Interface (GGSCI). For this, you need to identify the public IP address of the compute node where Oracle GoldenGate Classic for Big Data is running. The following steps help you to identify the public IP address:

- Log in to your Oracle Cloud Console.
- Navigate to **Compute -> Instances**.
- Select the hyperlink name of the compute node. The public IP address is listed under **Primary VNIC Information**.
- To access the compute node where Oracle GoldenGate is running, connect as the `opc` account using SSH. For more information on how to access a node using the `opc` account refer to [Connecting to an Instance](#).

A public SSH key is specified as a part of the deployment process and you can use the private key when you are connecting to the Oracle GoldenGate instance. The following example illustrates how you connect to the Oracle GoldenGate compute node:

```
ssh -i <private-key-filename> opc@<public-id-address>
```

Installing Third Party Drivers or Libraries

Oracle GoldenGate Classic for Big Data compute node has Oracle Java run time already installed and pre-configured. However, the Oracle GoldenGate Classic for Big Data compute node does not have the required third party drivers or libraries. You have to install these drivers or libraries before using Oracle GoldenGate for Big Data. You can identify the third party libraries based on the sources or targets that you wish to connect.

For example, before creating a kafka connection, you need to download all the dependent libraries from Kafka Maven repository depending on the Kafka version. The list of dependent libraries for each version of Kafka is listed in the [Kafka](#) section of *Using Oracle GoldenGate for Big data* guide.

For more information on dependent libraries, refer to [Using Oracle GoldenGate for Big Data](#) guide.

Running GGSCI

Oracle GoldenGate Classic for Big Data allows you to quickly access the GoldenGate Service Command Interface (GGSCI). After logging in to the compute node, you can find `GGSCI` from `home -> bigdata` directory (`/home/opc/bigdata`). From here, you can execute GGSCI.

To start GGSCI, execute the following commands:

```
$ cd bigdata
$ export LD_LIBRARY_PATH=/home/opc/java/jre/lib/amd64/server
$ ./ggsci
```

Working with Oracle GoldenGate Classic for Big Data

After deploying Oracle GoldenGate Classic for Big Data on Oracle Cloud Marketplace, you can access the latest release of Oracle GoldenGate Classic for Big Data.

Before you can start using Oracle GoldenGate Classic for Big Data, there are a few tasks that you must perform to ensure that your environment is complete and ready to replicate your data. Before you begin data replication, you must perform the following tasks:

Topics:

- [Configuring Source and Target Data System for Replication](#)
- [Establishing Connectivity](#)
- [Configuring Capture Support](#)
- [Configuring Apply Support](#)

Configuring Source and Target Data System for Replication

Before you can begin replicating data, you should prepare the source or target database to support Oracle GoldenGate. To prepare your database, follow the steps listed in [Installing Oracle GoldenGate for Big Data](#) in *Installing and Upgrading Oracle GoldenGate for Big Data* guide. See [Using the Generic Build of Oracle GoldenGate](#) to capture and/or apply data to Big Data and Cloud systems.

Establishing Connectivity

The Oracle GoldenGate Classic for Big Data on Oracle Cloud Marketplace compute node acts as a hub where you can manage your connections to source and/or target Big Data systems. To do this, establish a network connection between OCI compute node and your source and/or target Big Data systems. For this, you have to install the required third party connection libraries on the OCI compute node.

For more details on this, refer to [Installing Oracle GoldenGate for Big Data](#) chapter of *Installing and Upgrading Oracle GoldenGate for Big Data* guide.

Configuring Capture Support

Before you can begin replication, you have to setup the capture process. The capture process is also known as Extract.

Before installing Remote Extract for Cassandra source, refer to [Setting Up Cassandra Change Data Capture](#) section in *Using Oracle GoldenGate for Big Data* Guide, to know about the steps that you need to perform on the Cassandra platform for the replication to happen. For setting up JMS remote capture, refer to [Configuring Message Capture](#) section in *Administering Oracle GoldenGate for Big Data* guide.

To build the remote extract, perform the following steps within Oracle GoldenGate for Non-Oracle:

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Start GGSCI.

```
$ cd bigdata  
$ ggsci
```

3. Connect to the source database using the credential store.

```
ggsci > alter credentialstore add user <user_id>@<tnsnames_string>  
password ***** alias <alias_name> domain OracleGoldenGate
```

4. Add an Extract.

- Remote Extract

```
ggsci > add EXTRACT <extract_name>, VAM, begin now  
ggsci > add EXTTRAIL ./dirdat/id, EXTRACT <extract_name>, megabytes 100
```

Note

The `extract_name` parameter can be replaced with any process name that can be a maximum of 8 characters in length. The trail identifier (`id`) can be a maximum of two characters.

Configuring Apply Support

The apply process for replication, also known as Replicat, is very easy and simple to configure. To build a Replicat, perform the following steps with Oracle GoldenGate for Big Data.

1. Log in to the Oracle GoldenGate Compute Node.

```
$ ssh -i <private key> opc@<public_ip_address>
```

2. Start GGSCI.

```
$ cd bigdata  
$ export LD_LIBRARY_PATH=/home/opc/java/jre/lib/amd64/server  
$ ./ggsci
```

3. Add a Replicat.

```
ggsci > add REPLICAT <replicat_name>, EXTTRAIL ./dirdat/aa
```

Note

- You can replace `replicat_name` and `trail_name` characters specified above with any valid name.
- Process names can be a maximum of 8 characters and trail names can be two characters in length.

Connecting to Data Resources

Connecting to Big Data Systems

You can use Oracle GoldenGate Classic for Big Data on Oracle Cloud Marketplace to remotely capture and apply data to Big Data Database on-premises or across Oracle Cloud platforms. This allows you to enable replication between cloud services in a scalable fashion and centrally manage the replication processes between cloud services. To know more about the supported Big Data systems and their versions, refer to [GoldenGate for Big Data 19.1 Certification Matrix](#).

Use Cases for Replication

Use Oracle GoldenGate Classic for Big Data to replicate data between data resources in the following use cases:

- Data Migrations
- Data Distribution
- Real-Time Data Warehousing
- Operational Reporting
- Building Data Lakes
- Streaming Analytics

Prerequisites

Ensure that the following are set up before you begin replication:

- Oracle GoldenGate Classic for Big Data
- Any Sources and Targets that are supported by GoldenGate for BigData

To move data from on-premises to the cloud or from on-premises to on-premises, perform the tasks listed in the following sections:

- [Configuring Big Data Systems for Replication](#)
- [Configuring Oracle GoldenGate Classic for Big Data Compute Node](#)

Configuring Big Data Systems for Replication

To prepare your Big Data systems for replication, perform the tasks that are outlined in [Installing Oracle GoldenGate for Big Data](#) chapter of *Installing and Upgrading Oracle GoldenGate for Big Data* guide.

If your sources or targets are other data sources and to know about the various data sources supported by Oracle Golden Gate, refer to [Using Oracle GoldenGate for Oracle Database](#) and [Using Oracle GoldenGate for Heterogeneous Databases](#) guides.

Configuring Oracle GoldenGate Classic for Big Data Compute Node

To establish connection from Oracle GoldenGate Classic for Big Data compute node to your on-premises data systems or third party cloud systems, you must establish connectivity between OCI and on-premises data systems. To do this, you can use the following options:

- IPSec VPN
- FastConnect

For establishing an IPSec VPN with OCI, refer to [IPSec VPN](#) documentation. For establishing a FastConnect connection, refer to the [FastConnect](#) documentation.

Oracle GoldenGate Classic for SQL Server

Deploying Oracle GoldenGate Classic Architecture for SQL Server Platforms

Using Oracle GoldenGate Classic Architecture on Marketplace, you can deploy and manage your deployment from a single location.

Here are the prerequisites which are required before deployment:

- [Oracle Cloud Account](#)
- Access to compute node resources within Oracle Cloud Tenant

Finding Oracle GoldenGate for Non-Oracle (SQL Server) within the Oracle Cloud Marketplace

The following steps help you to deploy Oracle GoldenGate for Non-Oracle (SQL Server) from Oracle Cloud Marketplace:

1. Log in to [Oracle Cloud Marketplace](#).
2. From the Oracle Cloud Marketplace home page, use the search box under **Applications** and search for the keyword **GoldenGate**.
3. From the Search Results, select **Oracle GoldenGate for Non-Oracle**.

Launching Oracle GoldenGate Classic for Non-Oracle (SQL Server)

The following steps help you to launch Oracle GoldenGate Classic for Non-Oracle (SQL Server) on Oracle Cloud Marketplace:

1. From the application page, select **Get App**.
2. Select **OCI Region** or Log in using your OCI account login credentials.
 - a. OCI Region – Select the OCI Region for which you want to install the compute node.
 - b. Click **Sign In**.
3. In the **Sign In to the Cloud Tenant** page, provide the OCI tenant details.
You may be asked to sign in to your identity provider.
4. On the **Oracle GoldenGate for Non-Oracle (SQL Server)** page, find the selection box and provide the following information:
 - a. **Select Version** – It provides a list of versions that are available in the listing. It is set to Oracle GoldenGate for Non-Oracle (SQL Server) by default.

- b. **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.
 - c. **Terms of Use** – This check box is selected by default. Oracle recommends to review the licenses before proceeding with the instance creation.
 - d. **Launch Stack** – It launches the stack in the OCI environment.
5. 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 that you provide while creating the Stack.
 - Create In Compartment – It defaults to the compartment you have selected on the Oracle GoldenGate 19c for Oracle 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**.
6. Fill in the required details to **Configure Variables**. This information is required to build the compute node for for Non-Oracle (SQL Server):
 - a. Name for New Resources -
 - **Display Name** - Display Name used to identify all new OCI resources.
 - **Host DNS Name** – Domain Name Service for the new compute node.
 - b. Network Settings -
 - **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 such as:
 - Network Compartment
 - New VCN DNS Name
 - New VCN CIDR
 - New Subnet DNS Name
 - New Subnet CIDRIf 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.
 - **Network Compartment (optional)** – Compartment for new or existing network resources.
 - **VCN (optional)** – Existing VCN to use for the newly created instance if you are not creating a new network.
 - **Subnet (optional)** – Existing subnet to use for the newly created instance if you are not creating a new network. The subnet that you have selected must match the same Availability Domain set in the Instance Settings.
 - c. Instance Settings -
 - **Availability Domain** – It specifies the availability domain for the newly created Oracle GoldenGate Instance. It must match the Subnet that you have selected in the Use Existing Network settings.

- **Compute Shape** – Shape of new compute instance. Supported shapes are VM.Standard2.4, VM.Standard2.8, VM.Standard2.16, VM.Standard2.24
- **Assign Public IP** – Indicates if the new VM should have a public IP address.
- **Custom Volume Sizes** – Use this checkbox to customize the size of new block storage volumes.

- d. **SSH Public Key** - Public Key for allowing SSH access as the 'opc' user.

7. Click **Next**.
8. On the Review page, review the information you provided then click **Create**.
9. After clicking **Create**, you will be taken to the **Stacks** page. You can monitor the creation of the compute node using this page.
10. Upon completion, you can view the Oracle GoldenGate for Non-Oracle (SQL Server) compute node under **Compute -> Instances**.

Connecting to Oracle GoldenGate Classic for Non-Oracle (SQL Server) Compute Node

To access your new Oracle GoldenGate deployment, log in to the compute node and access Oracle GoldenGate Software Command Interface (GGSCI). For this, you need to identify the public IP address of the compute node where Oracle GoldenGate Classic is running. The following steps help you to identify the public IP address:

1. Log in to your Oracle Cloud Console.
2. Select **Compute -> Instances**.
3. Select the hyperlink name of the compute node. The public IP Address is listed under **Primary VNIC Information**.
4. To access the compute node where Oracle GoldenGate is running, connect as the `opc` account using SSH. For more information on how to access a node using the `opc` account refer to [Connecting to an Instance](#).

A public SSH key is specified as a part of the deployment process and you can use the private key when you are connecting to the Oracle GoldenGate instance. The following example illustrates how you connect to the Oracle GoldenGate compute node:

```
ssh -i <private-key-filename> opc@<public-id-address>
```

Working with Oracle GoldenGate Classic for Non-Oracle (SQL Server)

After deploying Oracle GoldenGate Classic for Non-Oracle (SQL Server) on Oracle Cloud Marketplace, you can access the latest release of Oracle GoldenGate Classic for Non-Oracle (SQL Server).

Before you can start using Oracle GoldenGate Classic for Non-Oracle (SQL Server), there are a few tasks that you must perform to ensure that your environment is complete and ready to replicate your data. Before you begin data replication, you must perform the following tasks:

Topics:

- [Establishing Connectivity](#)
- [Oracle GoldenGate for SQL Server Requirements](#)
- [Creating Database User and Schema](#)
- [Installing Microsoft ODBC Drivers for Linux](#)
- [Configuring ODBC Connectivity](#)
- [Creating GLOBALS File and Starting GGSCI](#)
- [Enabling Supplemental Logging for a Source SQL Server Database](#)
- [Configuring and Creating a Remote Extract for SQL Server](#)
- [Configuring and Creating a Remote Replicat for SQL Server](#)

Establishing Connectivity

The Oracle GoldenGate Classic for Non-Oracle (SQL Server) on Oracle Cloud Marketplace compute node acts as a hub where you can manage your connections to source and target SQL Server databases. To establish connectivity from an Oracle GoldenGate Classic for Non-Oracle (SQL Server) compute node to your on-premises SQL Server database, use the following options:

- IPSec VPN
- FastConnect

For establishing an IPSec VPN with OCI, refer to [IPSec VPN](#) documentation. For establishing a FastConnect connection, refer to the [FastConnect](#) documentation.

Oracle GoldenGate for SQL Server Requirements

Before you can begin replicating data, for a SQL Server database, review the instance, database and user requirements available in [Installing Oracle GoldenGate for SQL Server Databases](#) chapter of *Installing Oracle GoldenGate Guide* and [Using Oracle GoldenGate for SQL Server](#) chapter of *Using Oracle GoldenGate for Heterogeneous Databases*. These sections contain important information regarding database requirements and supported features.

Creating Database User and Schema

To support remote capture and apply for a SQL Server database, you must create a SQL Server login and schema.

For capturing data from a SQL Server source database, create a SQL Server login and schema that can be used by Oracle GoldenGate, and grant `sysadmin` access privileges for login. Oracle GoldenGate requires `sysadmin` access privileges to perform several prerequisites, but you can set the account to `dbowner` afterwards, which is required for the capture process to run. Create a schema in the database to be used as the source database.

Example for a Source Database :

```
CREATE LOGIN [oggsourceuser] WITH PASSWORD=N'password';
ALTER SERVER ROLE [sysadmin] ADD MEMBER [oggsourceuser];
USE [source_dbname];
CREATE SCHEMA [oggsourceschema];
```

If you are applying data to a SQL Server target database, create a SQL Server login that has dbowner access privileges within the target database, as well as a schema to be used for a Replicat's checkpoint table.

Example:

```
CREATE LOGIN [oggtargetuser] WITH PASSWORD=N'password';
USE [target_dbname];
ALTER ROLE [db_owner] ADD MEMBER [oggtargetuser];
CREATE SCHEMA [oggtargetschema];
```

Installing Microsoft ODBC Drivers for Linux

The Microsoft ODBC Drivers for Linux are required to connect to a remote source or target SQL Server database that runs on Windows. The following tasks are required to install the Linux drivers.

1. Edit the file `/etc/passwd`, to grant temporary shell access to the root user.

```
$ sudo vi /etc/passwd
```

2. In the file `/etc/passwd`, change the value for the root user from `/usr/sbin/nologin` to `/bin/bash`. Save and close the file.
3. Using Microsoft's RedHat Enterprise Server installation instructions for adding the ODBC Drivers for Linux, perform the following steps with default values by answering 'y' when prompted.

```
$ sudo su
```

```
$ #RedHat Enterprise Server 7
$ curl https://packages.microsoft.com/config/rhel/7/prod.repo > /etc/yum.repos.d/mssql-release.repo
```

```
$ exit
$ sudo yum remove unixODBC-utf16 unixODBC-utf16-devel #to avoid conflicts
$ sudo ACCEPT_EULA=Y yum install msodbcsql17
$ sudo ACCEPT_EULA=Y yum install mssql-tools
$ echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bash_profile
$ echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >> ~/.bashrc
$ source ~/.bashrc
```

4. After installing the Linux drivers, you can reset the original shell access values for the root user.

```
$ sudo vi /etc/passwd
```

5. Change the value for the root user from `/bin/bash` to `/usr/sbin/nologin`. Save and close the file.

Configuring ODBC Connectivity

Create and install an ODBC connection template file that contains the database connection string by using the login that you created for Oracle GoldenGate. Ensure that the TCP/IP

protocol is enabled for the SQL Server instance as Oracle GoldenGate will connect over TCP/IP for establishing the connection.

1. Create a template file for your data source:

```
$ vi odbc_template_file.ini
```

2. Describe the data source in the template file. In the following example, `myserver_sql2017_source` is used as the DSN name with `DBLOGIN` and `SOURCEDB/TARGETDB` to connect to the database.

```
[myserver_sql2017_source]
Driver = ODBC Driver 17 for SQL Server
Server = myserver,1433
Database = dbname
User = oggsOURCEuser
Password = password
```

Save and close the file.

3. Install the data source using the command.

```
$ odbcinst -i -s -f odbc_template_file.ini
```

Creating GLOBALS File and Starting GGSCI

Oracle GoldenGate Classic for Non-Oracle (SQL Server) allows you to quickly access the GoldenGate Service Command Interface (GGSCI) and is preconfigured with a running Manager process. After logging in to the compute node, you can find GGSCI from the `mssql` directory (`/home/opc/mssql`).

1. Create a `GLOBALS` file (with no extension) in the Oracle GoldenGate installation directory.

```
$ cd ~/mssql
$ vi GLOBALS
```

Add the following (minimum required) parameter to the `GLOBALS` file, substituting the correct value for the schema that you created earlier in the database:

```
GGSCHEMA oggsOURCEschema
```

Save and close the `GLOBALS` file.

2. To start GGSCI, execute the following command:

```
$ ./ggsci
```

Enabling Supplemental Logging for a Source SQL Server Database

You must enable the database and tables for SQL Server Change Data Capture, to capture transactional data from a source SQL Server database. Perform the following steps to enable database and table level Change Data Capture and install the Oracle GoldenGate CDC cleanup tasks.

1. From GGSCI, connect to the source database with a SQL Server login that has sysadmin access privileges.

Example:

```
GGSCI> DBLOGIN SOURCEDB myserver_sql2017_source USERID oggsourceuser  
PASSWORD password  
GGSCI> ADD TRANDATA dbo.table1  
GGSCI> ADD TRANDATA dbo.table2
```

2. Using SQL Server Management Studio, drop the SQL Server CDC Cleanup job by running the following against the database. Alternatively, you can manually disable cdc dbname_cleanup job within SQL Server Management Studio.

```
exec sys.sp_cdc_drop_job N'cleanup';
```

3. Execute the shell script to install the Oracle GoldenGate CDC cleanup tasks.

Syntax:

```
ogg_cdc_cleanup_setup.sh createJob userid password dbname server, port  
ggschema
```

Example:

```
GGSCI> shell ./ogg_cdc_cleanup_setup.sh createJob oggsourceuser password  
source_dbname myserver,1433 oggsourceschema
```

Configuring and Creating a Remote Extract for SQL Server

To capture transactional data from a source SQL Server database, you must configure a capture process, also known as an Extract. To build an Extract, you can perform the following steps with Oracle GoldenGate for Non-Oracle.

1. Create an Extract parameter file.

```
GGSCI> EDIT PARAMS extsql
```

Sample Extract parameter file:

```
EXTRACT extsql  
SOURCEDB myserver_sql2017_source USERID oggsourceuser PASSWORD password  
EXTTRAIL ./dirdat/et  
TABLE dbo.table1;  
TABLE dbo.table2;
```

Save and close the file.

2. Add the Extract and Extract's local trail.

① Note

While adding the Extract with the BEGIN NOW option, ensure to check if both the database server and the server running Oracle GoldenGate have the same system time.

```
GGSCI> ADD EXTRACT extsql, TRANLOG, BEGIN NOW
GGSCI> ADD EXTTRAIL ./dirdat/et, EXTRACT extsql
```

3. Verify the connectivity to the SQL Server source database and start the Extract.

```
GGSCI> DBLOGIN SOURCEDB myserver_sql2017_source USERID oggsourceuser
PASSWORD password
GGSCI> START EXTRACT extsql
```

Configuring and Creating a Remote Replicat for SQL Server

The apply process for replication, also known as Replicat, is used to deliver data to a target database. To build a Replicat, perform the following steps with Oracle GoldenGate for Non-Oracle.

1. From GGSCI, log in to the target database and create a checkpoint table.

```
GGSCI> DBLOGIN SOURCEDB myserver_sql2017_target USERID oggtargetuser
PASSWORD password
GGSCI> ADD CHECKPOINTTABLE oggtargetschema.oggcheck
```

2. Create a Replicat parameter file.

```
GGSCI > EDIT PARAMS repsql
```

Sample Replicat parameter file:

```
REPLICAT repsql
TARGETDB myserver_sql2017_target USERID oggtargetuser PASSWORD password
MAP dbo.table1, TARGET dbo.table1;
MAP dbo.table2, TARGET dbo.table2;
```

Save and close the file.

3. Add and start the Replicat.

```
GGSCI> ADD REPLICAT repsql, EXTTRAIL ./dirdat/et, CHECKPOINTTABLE
oggtargetschema.oggcheck
GGSCI> START REPLICAT repsql
```

Oracle GoldenGate Classic for PostgreSQL

Deploying Oracle GoldenGate Classic Architecture for PostgreSQL Platforms

Using Oracle GoldenGate Classic Architecture on Marketplace, you can deploy and manage your deployment from a single location.

Here are the prerequisites which are required before deployment:

- [Oracle Cloud Account](#)
- Access to compute node resources within Oracle Cloud Tenant

Finding Oracle GoldenGate for Non-Oracle (PostgreSQL) within the Oracle Cloud Marketplace

The following steps help you to deploy Oracle GoldenGate for Non-Oracle (PostgreSQL) from Oracle Cloud Marketplace:

1. Log in to [Oracle Cloud Marketplace](#).
2. From the Oracle Cloud Marketplace home page, use the search box under **Applications** and search for the keyword **GoldenGate**.
3. From the Search Results, select **Oracle GoldenGate for Non-Oracle**.

Launching Oracle GoldenGate Classic for Non-Oracle (PostgreSQL)

The following steps help you to launch Oracle GoldenGate Classic for Non-Oracle (PostgreSQL) on Oracle Cloud Marketplace:

1. From the application page, select **Get App**.
2. Select **OCI Region** or Log in using your OCI account login credentials.
 - a. OCI Region – Select the OCI Region for which you want to install the compute node.
 - b. Click **Sign In**.
3. In the **Sign In to the Cloud Tenant** page, provide the OCI tenant details.
You may be asked to sign in to your identity provider.
4. On the **Oracle GoldenGate for Non-Oracle (PostgreSQL)** page, find the selection box and provide the following information:
 - a. **Select Version** – It provides a list of versions that are available in the listing. It is set to Oracle GoldenGate for Non-Oracle (PostgreSQL) by default.

- b. **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.
 - c. **Terms of Use** – This check box is selected by default. Oracle recommends to review the licenses before proceeding with the instance creation.
 - d. **Launch Stack** – It launches the stack in the OCI environment.
5. 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 that you provide while creating the Stack.
 - Create In Compartment – It defaults to the compartment you have selected on the Oracle GoldenGate 19c for Oracle 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**.
6. Fill in the required details to **Configure Variables**. This information is required to build the compute node for for Non-Oracle (PostgreSQL):
 - a. Name for New Resources -
 - **Display Name** - Display Name used to identify all new OCI resources.
 - **Host DNS Name** – Domain Name Service for the new compute node.
 - b. Network Settings -
 - **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 such as:
 - Network Compartment
 - New VCN DNS Name
 - New VCN CIDR
 - New Subnet DNS Name
 - New Subnet CIDR
 - 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.
 - **Network Compartment** (optional) – Compartment for new or existing network resources.
 - **VCN** (optional) – Existing VCN to use for the newly created instance if you are not creating a new network.
 - **Subnet** (optional) – Existing subnet to use for the newly created instance if you are not creating a new network. The subnet that you have selected must match the same Availability Domain set in the Instance Settings.
 - c. Instance Settings -
 - **Availability Domain** – It specifies the availability domain for the newly created Oracle GoldenGate Instance. It must match the Subnet that you have selected in the Use Existing Network settings.

- **Compute Shape** – Shape of new compute instance. Supported shapes are VM.Standard2.4, VM.Standard2.8, VM.Standard2.16, VM.Standard2.24
- **Assign Public IP** – Indicates if the new VM should have a public IP address.
- **Custom Volume Sizes** – Use this checkbox to customize the size of new block storage volumes.

- SSH Public Key** - Public Key for allowing SSH access as the 'opc' user.

7. Click **Next**.
8. On the Review page, review the information you provided then click **Create**.
9. After clicking **Create**, you will be taken to the **Stacks** page. You can monitor the creation of the compute node using this page.
10. Upon completion, you can view the Oracle GoldenGate for Non-Oracle (PostgreSQL) compute node under **Compute -> Instances**.

Connecting to Oracle GoldenGate Classic for Non-Oracle (PostgreSQL) Compute Node

To access your new Oracle GoldenGate deployment, log in to the compute node and access Oracle GoldenGate Software Command Interface (GGSCI). For this, you need to identify the public IP address of the compute node where Oracle GoldenGate Classic is running. The following steps help you to identify the public IP address:

1. Log in to your Oracle Cloud Console.
2. Select **Compute -> Instances**.
3. Select the hyperlink name of the compute node. The public IP Address is listed under **Primary VNIC Information**.
4. To access the compute node where Oracle GoldenGate is running, connect as the `opc` account using SSH. For more information on how to access a node using the `opc` account refer to [Connecting to an Instance](#).

A public SSH key is specified as a part of the deployment process and you can use the private key when you are connecting to the Oracle GoldenGate instance. The following example illustrates how you connect to the Oracle GoldenGate compute node:

```
ssh -i <private-key-filename> opc@<public-id-address>
```

Working with Oracle GoldenGate Classic for Non-Oracle (PostgreSQL)

After deploying Oracle GoldenGate Classic for Non-Oracle (PostgreSQL) on Oracle Cloud Marketplace, you can access the latest release of Oracle GoldenGate Classic for Non-Oracle (PostgreSQL).

Before you can start using Oracle GoldenGate Classic for Non-Oracle (PostgreSQL), there are a few tasks that you must perform to ensure that your environment is complete and ready to replicate your data. Before you begin data replication, you must perform the following tasks:

Topics:

- [Establishing Connectivity](#)
- [Oracle GoldenGate for PostgreSQL Requirements](#)
- [Creating Database User and Schema](#)
- [Setting up and Configuring odbc.ini File](#)
- [Starting GGSCI](#)
- [Registering a Replication Slot](#)
- [Enabling Supplemental Logging for a Source PostgreSQL Database](#)
- [Configuring and Creating a Remote Extract for PostgreSQL](#)
- [Configuring and Creating a Remote Replicat for PostgreSQL](#)

Establishing Connectivity

The Oracle GoldenGate Classic for Non-Oracle (PostgreSQL) on Oracle Cloud Marketplace compute node acts as a hub where you can manage your connections to source and target PostgreSQL databases. To establish connectivity from an Oracle GoldenGate Classic for Non-Oracle (PostgreSQL) compute node to your on-premises PostgreSQL database, use the following options:

- IPSec VPN
- FastConnect

For establishing an IPSec VPN with OCI, refer to [IPSec VPN](#) documentation. For establishing a FastConnect connection, refer to the [FastConnect](#) documentation.

Oracle GoldenGate for PostgreSQL Requirements

Before you can begin replicating data, for a PostgreSQL database, review the instance, database, `LD_LIBRARY_PATH`, and user requirements.

To replicate data:

1. Oracle GoldenGate for PostgreSQL requires certain PostgreSQL client libraries to replicate the data and therefore, you must install minimum PostgreSQL client library. See [Installing for PostgreSQL](#) in *Installing Oracle GoldenGate Guide* and [Using Oracle GoldenGate for PostgreSQL](#) chapter of *Using Oracle GoldenGate for Heterogeneous Databases*. These sections contain important information regarding database requirements and supported features.
 - a. To install PostgreSQL client, execute: `sudo yum install postgresql-client`.

Note

While installing PostgreSQL client or plugin, ensure that the PostgreSQL database version matches with PostgreSQL client and plugin.

- For database version PostgreSQL 11, it is `sudo yum install postgresql11-contrib`.
- For database version PostgreSQL 10, it is `sudo yum install postgresql10-contrib`.

2. Check whether or not the PostgreSQL client library (`LIBPQ`) exists.
3. After the installation is complete, add the `LD_LIBRARY_PATH`. If the PostgreSQL client library path is `-/opt/pg/lib` and the Oracle GoldenGate home directory path is `/opt/ogg/postgresql`, then set `LD_LIBRARY_PATH` as `export LD_LIBRARY_PATH= /opt/pg/lib:/opt/ogg/postgresql/lib:$LD_LIBRARY_PATH`.

Creating Database User and Schema

To create database user and schema:

1. Create database user `ogg1` with password `ogg1`. See [PostgreSQL 12 Documentation - Create User](#).
2. Create schema. See [PostgreSQL 12 Documentation - Create Schema](#).
3. Create database. See [PostgreSQL 12 Documentation - Create Database](#).
4. Grant role permissions to the user for more privileges. See [PostgreSQL 12 Documentation - Grant Role](#).

```
create user ogg1 with password 'ogg1';
CREATE SCHEMA dbo;
CREATE DATABASE qadb;
GRANT replication ON DATABASE qadb TO ogg1;
```

Setting up and Configuring odbc.ini File

For more information about setting up `odbc.ini` and `LD_LIBRARY_PATH`, see Database Requirements PostgreSQL in the *Installing Oracle GoldenGate* Guide.

Starting GGSCI

Oracle GoldenGate Classic for Non-Oracle (PostgreSQL) allows you to quickly access the GoldenGate Service Command Interface (GGSCI) and is preconfigured with a running Manager process. After logging in to the compute node, you can find GGSCI from the `postgres` directory (`/home/opc/postgres`).

To start GGSCI:

1. Run the following command:

```
export LD_LIBRARY_PATH=/home/opc/postgres/lib:/opt/PostgreSQL/pg11/
lib:$LD_LIBRARY_PATH
```

2. To start GGSCI, execute the following command:

```
$ ./ggsci
```

Registering a Replication Slot

Oracle GoldenGate needs to register the extract with the database replication slot.

Before registering the Extract, ensure to have the DBLOGIN connected to the source database.

For example, if the Extract group name is `extr`, then run the following command from GGSCI to register the Extract:

```
GGSCI> register extract extr with <dbname>
```

Enabling Supplemental Logging for a Source PostgreSQL Database

You must enable the supplemental logging of tables for PostgreSQL Change Data Capture, to capture transactional data from a source PostgreSQL database. Perform the following steps to enable table level Change Data Capture.

To enable supplemental logging for a source PostgreSQL:

- From GGSCI, connect to the source database with a PostgreSQL login:

```
GGSCI> DBLOGIN sourcedb <DSN> USERID <oggsourceuser> PASSWORD <Pwd>
GGSCI > ADD TRANDATA dbo.table1
GGSCI > ADD TRANDATA dbo.table2
```

Configuring and Creating a Remote Extract for PostgreSQL

To capture transactional data from a source PostgreSQL database, you must configure a capture process, also known as an Extract. To build an Extract, you can perform the following steps with Oracle GoldenGate for Non-Oracle.

For more information about connecting to Remote Capture and Apply, see How to Connect Remote Capture and Apply for PostgreSQL in *Using Oracle GoldenGate for Heterogeneous Databases* Guide.

- Create an Extract parameter file.

```
GGSCI> EDIT PARAMS extsql
```

Sample Extract parameter file:

```
EXTRACT extsql
DBLOGIN sourcedb <DSN> USERID <oggsourceuser> PASSWORD <Pwd>
EXTTRAIL ./dirdat/et
TABLE dbo.table1;
TABLE dbo.table2;
```

Save and close the file.

- Add the Extract and Extract's local trail.

```
GGSCI> ADD EXTRACT extsql, TRANLOG, BEGIN NOW
GGSCI> ADD EXTTRAIL ./dirdat/et, EXTRACT extsql
```

- Verify the connectivity to the PostgreSQL source database and start the Extract.

```
GGSCI> DBLOGIN sourcedb <DSN> USERID <oggsourceuser> PASSWORD <Pwd>
GGSCI> START EXTRACT extsql
```

Configuring and Creating a Remote Replicat for PostgreSQL

The apply process for replication, also known as Replicat, is used to deliver data to a target database. To build a Replicat, perform the following steps with Oracle GoldenGate for Non-Oracle.

1. From GGSCI, log in to the target database and create a checkpoint table.

```
GGSCI> DBLOGIN sourcedb <DSN> USERID <oggsourceuser> PASSWORD <Pwd>
GGSCI> ADD CHECKPOINTTABLE oggtargetschema.oggcheck
```

2. Create a Replicat parameter file.

```
GGSCI > EDIT PARAMS repsql
```

Sample Replicat parameter file:

```
REPLICAT repsql
TARGETDB <DSN> USERID <oggtargetuser> PASSWORD <Pwd>
MAP dbo.table1, TARGET dbo.table1;
MAP dbo.table2, TARGET dbo.table2;
```

Save and close the file.

3. Add and start the Replicat.

```
GGSCI> ADD REPLICAT repsql, EXTTRAIL ./dirdat/et, CHECKPOINTTABLE
oggtargetschema.oggcheck
GGSCI> START REPLICAT repsql
```

Oracle GoldenGate Classic for MySQL

Deploying Oracle GoldenGate Classic Architecture for MySQL Platforms

Using Oracle GoldenGate Classic Architecture on Marketplace, you can deploy and manage your deployment from a single location.

Here are the prerequisites which are required before deployment:

- [Oracle Cloud Account](#)
- Access to compute node resources within Oracle Cloud Tenant

Finding Oracle GoldenGate for Non-Oracle (MySQL) within the Oracle Cloud Marketplace

The following steps help you to deploy Oracle GoldenGate for Non-Oracle (MySQL) from Oracle Cloud Marketplace:

1. Log in to [Oracle Cloud Marketplace](#).
2. From the Oracle Cloud Marketplace home page, use the search box under **Applications** and search for the keyword **GoldenGate**.
3. From the Search Results, select **Oracle GoldenGate for Non-Oracle**.

Launching Oracle GoldenGate Classic for Non-Oracle (MySQL)

The following steps help you to launch Oracle GoldenGate Classic for Non-Oracle (MySQL) on Oracle Cloud Marketplace:

1. From the application page, select **Get App**.
2. Select **OCI Region** or Log in using your OCI account login credentials.
 - a. OCI Region – Select the OCI Region for which you want to install the compute node.
 - b. Click **Sign In**.
3. In the **Sign In to the Cloud Tenant** page, provide the OCI tenant details.
You may be asked to sign in to your identity provider.
4. On the **Oracle GoldenGate for Non-Oracle (MySQL)** page, find the selection box and provide the following information:
 - a. **Select Version** – It provides a list of versions that are available in the listing. It is set to Oracle GoldenGate for Non-Oracle (MySQL) by default.

- b. **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.
 - c. **Terms of Use** – This check box is selected by default. Oracle recommends to review the licenses before proceeding with the instance creation.
 - d. **Launch Stack** – It launches the stack in the OCI environment.
5. 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 that you provide while creating the Stack.
 - Create In Compartment – It defaults to the compartment you have selected on the Oracle GoldenGate 19c for Oracle 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**.
6. Fill in the required details to **Configure Variables**. This information is required to build the compute node for for Non-Oracle (MySQL):
 - a. Name for New Resources -
 - **Display Name** - Display Name used to identify all new OCI resources.
 - **Host DNS Name** – Domain Name Service for the new compute node.
 - b. Network Settings -
 - **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 such as:
 - Network Compartment
 - New VCN DNS Name
 - New VCN CIDR
 - New Subnet DNS Name
 - New Subnet CIDR
 - c. Instance Settings -
 - **Availability Domain** – It specifies the availability domain for the newly created Oracle GoldenGate Instance. It must match the Subnet that you have selected in the Use Existing Network settings.

- **Compute Shape** – Shape of new compute instance. Supported shapes are VM.Standard2.4, VM.Standard2.8, VM.Standard2.16, VM.Standard2.24
- **Assign Public IP** – Indicates if the new VM should have a public IP address.
- **Custom Volume Sizes** – Use this checkbox to customize the size of new block storage volumes.

- SSH Public Key** - Public Key for allowing SSH access as the 'opc' user.

7. Click **Next**.
8. On the Review page, review the information you provided then click **Create**.
9. After clicking **Create**, you will be taken to the **Stacks** page. You can monitor the creation of the compute node using this page.
10. Upon completion, you can view the Oracle GoldenGate for Non-Oracle (MySQL) compute node under **Compute -> Instances**.

Connecting to Oracle GoldenGate Classic for Non-Oracle (MySQL) Compute Node

To access your new Oracle GoldenGate deployment, log in to the compute node and access Oracle GoldenGate Software Command Interface (GGSCI). For this, you need to identify the public IP address of the compute node where Oracle GoldenGate Classic for Non-Oracle (MySQL) is running. The following steps help you to identify the public IP address:

1. Log in to your Oracle Cloud Console.
2. Select **Compute -> Instances**.
3. Select the hyperlink name of the compute node. The public IP Address is listed under **Primary VNIC Information**.
4. To access the compute node where Oracle GoldenGate is running, connect as the `opc` account using SSH. For more information on how to access a node using the `opc` account refer to [Connecting to an Instance](#).

A public SSH key is specified as a part of the deployment process and you can use the private key when you are connecting to the Oracle GoldenGate instance. The following example illustrates how you connect to the Oracle GoldenGate compute node:

```
ssh -i <private-key-filename> opc@<public-id-address>
```

Working with Oracle GoldenGate Classic for Non-Oracle (MySQL)

After deploying Oracle GoldenGate Classic for Non-Oracle (MySQL) on Oracle Cloud Marketplace, you can access the latest release of Oracle GoldenGate Classic for Non-Oracle (MySQL).

Before you can start using Oracle GoldenGate Classic for Non-Oracle (MySQL), there are a few tasks that you must perform to ensure that your environment is complete and ready to replicate your data.

Before you begin data replication, you must perform the following tasks:

Topics:

- [Establishing Connectivity](#)
- [Oracle GoldenGate for MySQL Requirements](#)
- [Configure Source MySQL Database for Remote Capture](#)
- [Configure Target MySQL Database for Remote Apply](#)
- [Starting GGSCI](#)
- [Configuring and Creating a Remote Extract for MySQL](#)
- [Configuring and Creating a Remote Replicat for MySQL](#)

Establishing Connectivity

The Oracle GoldenGate Classic for Non-Oracle (MySQL) on Oracle Cloud Marketplace compute node acts as a hub where you can manage your connections to source and target MySQL databases. To establish connectivity from an Oracle GoldenGate Classic for Non-Oracle (MySQL) compute node to your on-premises MySQL database, use the following options:

- IPSec VPN
- FastConnect

For establishing an IPSec VPN with OCI, refer to [IPSec VPN](#) documentation. For establishing a FastConnect connection, refer to the [FastConnect](#) documentation.

Oracle GoldenGate for MySQL Requirements

Before you can begin replicating data, for a MySQL database, review the instance, database and user requirements available in [Installing Oracle GoldenGate for MySQL](#) chapter of *Installing Oracle GoldenGate Guide* and [Understanding What's Supported for MySQL](#) chapter of *Using Oracle GoldenGate for Heterogeneous Databases*. These sections contain important information regarding database requirements and supported features.

Configure Source MySQL Database for Remote Capture

To support remote capture for MySQL database, configure the MySQL database as follows:

1. Create and grant access permissions to an Oracle GoldenGate remote capture user. Execute the following statements in a remote database to create a user and grant the required permissions for remote capture. The following example provides high level privileges for the created remote capture user :

```
mysql > CREATE USER 'sourceuser'@'dbhostname' IDENTIFIED BY 'password';
mysql > GRANT ALL PRIVILEGES ON *.* TO 'sourceuser'@'dbhostname';
mysql > FLUSH PRIVILEGES;
```

To know more about user permission requirements, refer to [Database User for Oracle GoldenGate Processes for MySQL](#) chapter of the *Using Oracle GoldenGate for Heterogeneous Databases* guide.

2. The `server_id` value of the remote MySQL server should be greater than 0. You can verify this value by executing the following statement on the MySQL remote server:

```
mysql > show variables like 'server_id';
```

If the `server_id` value is 0, modify the `my.cnf` configuration file to set it to a value greater than 0.

Configure Target MySQL Database for Remote Apply

To support remote delivery for MySQL database, create a remote apply user for the Oracle GoldenGate Replicat:

Create and grant access permissions to an Oracle GoldenGate remote apply user. Execute the following statements in a remote database to create a user and grant the required permissions for remote apply. The following example provides high level privileges for the created remote apply user :

```
mysql > CREATE USER 'targetuser'@'dbhostname' IDENTIFIED BY 'password';
mysql > GRANT ALL PRIVILEGES ON *.* TO 'targetuser'@'dbhostname';
mysql > FLUSH PRIVILEGES;
```

To know more about user permission requirements, refer to [Database User for Oracle GoldenGate Processes for MySQL](#) chapter of the *Using Oracle GoldenGate for Heterogeneous Databases* guide.

Starting GGSCI

Oracle GoldenGate Classic for Non-Oracle (MySQL) allows you to quickly access the GoldenGate Service Command Interface (GGSCI) and is preconfigured with a running Manager process. After logging in to the compute node, you can find GGSCI from the `mysql` directory (`/home/opc/mysql`).

To start GGSCI, execute the following command:

```
$ cd mysql
$ ./ggsci
```

Configuring and Creating a Remote Extract for MySQL

To capture transactional data from a source MySQL database, you must configure a capture process, also known as an Extract. To build an Extract, you can perform the following steps with Oracle GoldenGate for Non-Oracle.

1. Create an Extract parameter file.

```
GGSCI> EDIT PARAMS extmysql
```

Sample Extract parameter file:

```
EXTRACT extmysql
SOURCEDB sourcedb@dbhostname:port, USERIDALIAS name
TRANLOGOPTIONS ALTLOGDEST REMOTE
EXTTRAIL ./dirdat/et
TABLE sourcedb.*;
```

Save and close the file.

2. Add the Extract and Extract's local trail.

① Note

While adding the Extract with the BEGIN NOW option, ensure to check if, both the database server and the server running Oracle GoldenGate have the same system time.

```
GGSCI> ADD EXTRACT extmysql, TRANLOG, BEGIN NOW
GGSCI> ADD EXTTRAIL ./dirdat/et, EXTRACT extmysql
```

3. Verify the connectivity to MySQL source database and start the Extract.

```
GGSCI> DBLOGIN SOURCEDB sourcedb@dbhostname:port USERIDALIAS name
GGSCI> START EXTRACT extmysql
```

Configuring and Creating a Remote Replicat for MySQL

The apply process for replication, also known as Replicat, is used to deliver data to a target database. To build a Replicat, perform the following steps with Oracle GoldenGate for Non-Oracle.

1. From GGSCI, log in to the target database and create a checkpoint table.

```
GGSCI> DBLOGIN targetdb@dbhostname:port, USERIDALIAS name
GGSCI> ADD CHECKPOINTTABLE targetdb.ggcheck
```

2. Create a Replicat parameter file.

```
GGSCI > EDIT PARAMS repsql
```

Sample Replicat parameter file:

```
REPLICAT repmysql
TARGETDB targetdb@dbhostname:port, USERIDALIAS name
MAP sourcedb.table1, TARGET targetdb.table1;
MAP sourcedb.table2, TARGET targetdb.table2;
```

Save and close the file.

3. Add and start the Replicat.

```
GGSCI> ADD REPLICAT repmysql, EXTTRAIL ./dirdat/et, CHECKPOINTTABLE
targetdb.ggcheck
GGSCI> START REPLICAT repmysql
```

Part IV

Additional Topics

This part contains the following chapters:

- [Upscaling and Downscaling Oracle GoldenGate Marketplace Compute Shape Sizes](#)
- [Migrating Oracle GoldenGate Cloud Service \(Gen 1\) DIPC to Oracle GoldenGate Classic on Oracle Cloud Marketplace](#)
- [Upgrading Oracle GoldenGate Marketplace Image](#)
- [Manually Patching Oracle GoldenGate for Big Data Instance on Oracle Cloud Marketplace](#)
- [Removing Oracle GoldenGate Instance](#)
- [Troubleshooting](#)

Manually Patching Oracle GoldenGate for Big Data Instance on Oracle Cloud Marketplace

This chapter describes how to manually patch Oracle GoldenGate for Big Data instance 19c (19.1.0.0) image by downloading the product as described in [Downloading Oracle GoldenGate for Big Data](#), and then choosing the upgrade paths that suits your environment.

Note

The below procedure helps you to manually apply additional bug fixes on your existing marketplace place image. In the future if, you decide to upgrade your new marketplace image (as described in [Upgrading an Oracle Cloud Marketplace Image to 1.5.x \(Same Terraform Version\)](#) chapter) then make sure to manually verify all the fixes that you had previously applied on your existing marketplace image. Otherwise, you may miss those fixes by switching to image upgrade procedure.

Upgrading by Overwriting the Existing Installation

To upgrade your existing Oracle GoldenGate for Big Data marketplace image, copy the Oracle GoldenGate for Big Data 19c (19.1.0.0.x) files into the existing 19c (19.1.0.0.x) installation directory. Overwriting the product files is possible because there is neither structural nor package name changes in 19c (19.1.0.0.x).

1. (Source systems) Back up the current Oracle GoldenGate for Big Data installation directory on the source system, and any working directories that you have installed on a shared drive in a cluster (if applicable).
2. (Source system) Stop user activity on objects in the Oracle GoldenGate configuration.
3. Stop any running Extract and Replicat processes.
4. Stop the Manager process.

5. Verify whether all the above processes are stopped by using the command -

```
ggsci> info all
```

① Note

Make sure all the process are stopped before proceeding with the upgradation process.

6. Copy the new Oracle GoldenGate for Big Data 19c (19.1.0.0.x) zip to the location /u01/app/ogg/bigdata.

7. Navigate to the path /u01/app/ogg/bigdata using the command -

```
cd /u01/app/ogg/bigdata
```

8. Extract the new Oracle GoldenGate for Big Data 19c (19.1.0.0.x) zip file, which divides the files into several subdirectories.

① Note

Please note to replace x with appropriate patch version number in the below commands.

```
/u01/app/ogg/bigdata > unzip OGG_BigData_Linux_x64_19.1.0.0.x.zip  
/u01/app/ogg/bigdata > tar -vf OGG_BigData_Linux_x64_19.1.0.0.x.tar
```

9. Verify whether all the binary files got overridden by the command

```
/u01/app/ogg/bigdata > ls -ltr
```

10. Execute the command ggsci -v to verify the version upgrade.

11. Start the manager and Replicat processes and verify that they are running.

```
GGSCI> START MANAGER  
GGSCI> START REPLICAT group_name  
GGSCI> INFO REPLICAT group_name  
GGSCI> VIEW REPORT group_name
```

Upscaling and Downscaling Oracle GoldenGate Marketplace Compute Shape Sizes

This chapter helps you in dynamically changing the compute shape sizes of your Oracle GoldenGate on Oracle Cloud Marketplace compute node.

Upscaling an Oracle GoldenGate Compute Shape

To upscale your Oracle GoldenGate Compute Shape, perform the following tasks:

1. Login to your OCI Tenancy.
2. From the **Context** menu present on the top left corner of your screen, select **Compute -> Instances**.
3. Select the Oracle GoldenGate Compute Node that you wish to upscale.
4. From the **Instance Details** screen, locate the **Actions** drop down menu.
5. From the **Actions** drop down menu, select the option **Change Shape**. A list of available shapes appear, allowing you to select the desired shape for your selected compute instance.

Note

You may see more shapes listed, depending on your access to OCI compute resources. But you can only upscale to shapes that are supported for Oracle GoldenGate on Oracle Cloud Marketplace. For more information on this, refer to [Compute Shapes](#).

6. Select the required compute shape and click **Change Shape**.

This restarts your compute node. Once the compute node is restarted, check your access to ServiceManager by using the Public IP address.

Downscaling an Oracle GoldenGate Compute Shape

To downscale your Oracle GoldenGate Compute Shape, perform the following tasks:

1. Login to your OCI Tenancy.
2. From the **Context** menu present on the top left corner of your screen, select **Compute -> Instances**.
3. Select the Oracle GoldenGate Compute Node that you wish to downscale.
4. From the **Instance Details** screen, locate the **Actions** drop down menu.
5. From the **Actions** drop down menu, select the option **Change Shape**. A list of available shapes appear, allowing you to select the desired shape for your selected compute instance.

① Note

You may see more shapes listed, depending on your access to OCI compute resources. But you can only downscale to shapes that are supported for Oracle GoldenGate on Oracle Cloud Marketplace. For more information on this, refer to [Compute Shapes](#).

6. Select the required compute shape and click **Change Shape**.

This restarts your compute node. Once the compute node is restarted, check your access to ServiceManager by using the Public IP address.

Migrating Oracle GoldenGate Cloud Service (Gen 1) DIPC to Oracle GoldenGate Classic on Oracle Cloud Marketplace

Oracle GoldenGate on Oracle Cloud Marketplace is the customer managed replacement for Oracle GoldenGate Cloud Service (Gen 1). Oracle recommends you to move from Oracle GoldenGate Cloud Service (Gen 1) DIPC to Oracle GoldenGate Classic on Oracle Cloud Marketplace as soon as possible.

Refer to [Migrating Oracle GoldenGate Cloud Service to Oracle Cloud Marketplace - Classic Architecture](#), to migrate from Oracle GoldenGate Cloud Service (Gen 1) to Oracle GoldenGate Classic on Oracle Cloud Marketplace, .

Refer to [Migrate Your Data Integration Platform Cloud Classic Instances to Oracle Cloud Infrastructure](#), to migrate your Data Integration Platform Cloud Classic instances to Oracle Cloud Marketplace.

11

Upgrading Oracle GoldenGate Marketplace Image

This chapter helps you to upgrade an existing (or older version of) Oracle GoldenGate Marketplace stack to the latest release, using the OCI Stacks approach.

 **Note**

Upgrade from a 19c stack to 21c release is not permitted.

Before any upgrade, ensure to do the following:

1. Shut down all the Oracle GoldenGate processes in the VM instance.
2. Backup all the block storage just in case you need revert it back in the future for a failed upgrade/update.

Topics:

Stacks

Stacks are zip files that contain the latest Terraform code base. These 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.

When Stacks are available, you can access them through the Oracle Object Store.

Retrieving a Stack

Before you upgrade Oracle GoldenGate Marketplace, you have to retrieve the latest stack.

To retrieve a stack:

1. Go to the Market Place List.
2. Select a version from the Version drop-down list. This is the Market Place image you will use to upgrade.
3. Select the **I have reviewed and accepted the Oracle Program Limited Use Promotional Terms** check box.
4. Click **Download** to download the stack zip file and save it on your local drive in the following directory: `BundlePatch19c`.

Identifying the Terraform Version for the Stack to be Upgraded

To identify the Terraform version for the stack to be upgraded:

1. Log in to the Oracle Cloud Interface (OCI) Console.

2. On the Home page, under **Developer Services**, click **Resource Manager**, and then click **Stacks**.
3. Select an existing stack for which, you want to identify the Terraform version. Note that under **Stack Information**, the **Terraform version** field lists the Terraform version used by the stack.

 **Note**

If Terraform version of stack to be upgraded and the new stack version are same, then follow the steps in [Upgrading an Oracle Cloud Marketplace Image to 1.5.x \(Same Terraform Version\)](#). If the versions are different, follow the steps in [Upgrading an Oracle Cloud Marketplace Image \(To Different Versions of Terraform\)](#).

Upgrading an Oracle Cloud Marketplace Image (To Different Versions of Terraform)

To upgrade an Oracle Cloud Marketplace image, where the pre-upgrade stack Terraform version is updated to a different Terraform version in the new stack, perform the following tasks:

Verify that the Stack you want to Upgrade is up to Date

1. Verify that the stack you want to Upgrade is up to date.
2. In the left panel, select the compartment containing the stack you need to upgrade.
3. In the right panel, click the stack you need to upgrade.
4. In the **Stack details** page, make a note of the variables that the Stack used. In the left panel, click **Variables** and copy the variables into a text editor. This is required because you need to use the same compartment, vcn, or subnet eventually.
5. Make a note of the tags your Stack uses. In the right panel switch to the Tags pane and copy the tags into a text editor.
6. Confirm there are no pending changes to the stack by clicking the Plan operation on the Terraform Actions menu.

The job output must display **No changes. Infrastructure is up-to-date**.

Download the Terraform Configuration and Terraform State File from the Old Stack and Create a New Stack

1. Download the stack's Terraform Configuration and Terraform state file from your old version stack (rename the state file to `terraform.tfstate`). For details on downloading a stack's state file and Terraform configuration, see [Getting a Stack's State File](#) and [Downloading a Terraform Configuration File](#).
2. Copy the Terraform Configuration and Terraform state file into a local workspace and unzip the file.
3. Remove the zip file named `ocid*` after unzipping.

4. Use Terraform_13 (download from hashicorp and save to local host PATH variable) to upgrade the stack's Terraform Configuration.

Before running the upgrade, check the `main.tf` file located in the stack for the required Terraform version. If the Terraform required version is defined as `required_version = "~> 0.12"`, change it to `required_version = ">= 0.12"`.

5. Run the upgrade command on the bash prompt, as shown:

```
bash-4.2$ terraform_13 0.13upgrade
```

6. Create a new zip file for the updated stack configuration, as shown in the following code snippet:

```
bash-4.2$ zip -qr upgradeTF013_Oracle1923.zip *
```

In this snippet, `TF013_Oracle1923.zip` is the name of the new zip file.

Upgrade from Terraform v0.12 to v1.0.x

Upgrading the Oracle GoldenGate on Marketplace with Terraform v0.12 to 1.0.x is a three-part process:

1. Upgrade the Terraform stack from 0.12 to 0.13
2. Upgrade the Terraform stack from 0.13 to 0.14
3. Upgrade the Terraform stack from 0.14 to 1.0.x

Use the following instructions to upgrade from Terraform v 0.12 to v0.13, then from v0.13 to v0.14, and then 0.14 to 1.0.x.

1. Log in to the Oracle Cloud Interface (OCI) Console.
2. Login to your Oracle GoldenGate Instance and stop all the running Oracle GoldenGate processes. Ensure that you stop all the Microservices (Administration Service, Distribution Service, Receiver Service and Performance Metrics Service) including Service Manager.
3. On the Home page, click the navigation menu in the top left corner and under **Developer Services**, click **Resource Manager**, and then click **Stacks**.
4. Choose the new Stack you created in [Download the Terraform Configuration and Terraform State File from the Old Stack and Create a New Stack](#).
5. Click **Edit** from the top menu.
6. Upload the new zip (`terraform_13 upgraded zip`).
7. Update Terraform version to 0.13.x and click **Save Changes**.
8. Import the Terraform state file `terraform.tfstate` to the stack using **Import Job**.
9. Go to **Compute** and select **Instances**.
10. Terminate the existing compute instance that needs to be upgraded.
11. Ensure that the **Permanently delete the attached boot volume** option is selected.
12. After the instance is terminated, from the left navigation pane, select **Developer Services** and click **Resource Manager** and then click **Stacks**.
13. Select the Stack that you edited in step 4, for upgrade.
14. Click **Apply Job**. Ensure the step run without errors.

Wait until the job completes and verify whether the compute node is up and running.

15. To upgrade from Terraform v0.13 to v.14:

- a. From the Resource Manager, select the stack that you upgraded to 0.13 and click **Edit**.
- b. Update the Terraform version in the stack to 0.14 and click **Next**.
- c. Verify that the pre-filled details are accurate and click **Save Changes**.
- d. Go to **Compute** and select **Instances**.
- e. Terminate the existing compute instance that needs to be upgraded.
- f. Ensure that the **Permanently delete the attached boot volume** option is selected.
- g. After the instance is terminated, from the left navigation pane, select **Developer Services** and click **Resource Manager** and then click **Stacks**.
- h. Select the stack for which the Terraform version is to be upgraded.
- i. Click **Apply Job**. Wait until the job completes and verify whether the compute node is up and running

16. After successfully upgrading to Terraform v0.14, follow the steps provided in [Upgrading an Oracle Cloud Marketplace Image to 1.5.x \(Same Terraform Version\)](#).

17. After upgrade to Terraform v1.0.0.x completes successfully, remove the compute instance's IP address from the `~/.ssh/known_hosts` file, so that you can log in to the compute node to start all the Oracle GoldenGate processes.

Upgrading an Oracle Cloud Marketplace Image to 1.5.x (Same Terraform Version)

The upgrade process is only valid for the upgrade or patching within the same major version, for example, from v19c to v19c. Current Marketplace doesn't support major version upgrade through image update.

To upgrade an existing Oracle GoldenGate Marketplace image when the pre-upgrade stack Terraform version is the same as the new stack version, perform the following tasks:

Note

These steps are applicable in Oracle GoldenGate Marketplace that are updated from Terraform v 0.14 to 1.5.x only.

1. Download the correct version of the Stack.
2. Log in to the Oracle Cloud Interface (OCI) Console.
3. Login to your Oracle GoldenGate Instance and stop all the running Oracle GoldenGate processes. Ensure that you stop all the Microservices (Administration Service, Distribution Service, Receiver Service and Performance Metrics Service) including Service Manager.
4. On the Home page, click the navigation menu in the top left corner and under **Developer Services**, click **Resource Manager**, and then click **Stacks**.
5. Select an existing stack that you want to upgrade.
6. Click **Edit** from the top menu.
7. Click **Browse** to select the downloaded stack.
8. Upload the new .zip file.

9. Update the terraform version in the stack to 1.5.x and click **Next**.
10. Verify that the pre-filled details are accurate and click **Save Changes**.
11. Go to **Compute** and select **Instances**.
12. Terminate the existing compute instance that needs to be upgraded.
13. Ensure that the **Permanently delete the attached boot volume** option is selected.
14. After the instance is terminated, from the left navigation pane, select **Developer Services** and click **Resource Manager** and then click **Stacks**.
15. Select the **Stack** that you edited in step 6, for upgrade.
16. Click **Apply Job**.

Wait until the job completes and verify whether the compute node is up and running.

17. After upgrade, remove the compute instance's IP address from the `~/.ssh/known_hosts` file, so that you can log in to the compute node to start all the Oracle GoldenGate processes.

12

Removing Oracle GoldenGate Instance

To remove the Oracle GoldenGate Instance from your Oracle Cloud environment, follow these steps:

1. Log in to your Oracle Cloud Account.
2. Select Resource Manager → Stacks from the menu.
You can get a list of stacks that are built in your compartment.
3. Click the link of the stack that you want to remove.
4. In the Stack, select Destroy from the TerraForm Action drop-down list.
This permanently removes your Oracle GoldenGate Instance.
5. Delete the stack (optional).

OCI Policies Needed for Creating Oracle GoldenGate Marketplace Instances

The following OCI policies are required for creating Oracle GoldenGate Market Place instances:

- Allow group <ggowner> to manage *instance-family* in compartment <Compartmet Name>
- Allow group <ggowner> to manage *orm-family* in compartment <Compartmet Name>
- Allow group <ggowner> to manage *volume-family* in compartment <Compartmet Name>
- Allow group <ggowner> to use *virtual-network-family* in compartment <Compartmet Name>
- Allow group <ggowner> to manage *public-ips* in compartment <Compartmet 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 <Compartmet 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 manage public-ips permission is required only if you give the instance a public IP address. Oracle GoldenGate uses reserved IP addresses so that the public address is preserved across stack upgrades.
- 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>.

14

Troubleshooting

This chapter helps you in troubleshooting the problems that occur when you deploy Oracle GoldenGate Microservices on Oracle Cloud Marketplace.

Topics:

- [Logdump](#)
- [Enable Debug Logging for Oracle GoldenGate Microservices](#)

Logdump

The logdump utility is used to open, control the display, navigate through a file, search, filter, view and save data that is stored in the trail or extract file. This is a key utility in troubleshooting issues related to data within an Oracle GoldenGate environment.

For more information on how to use Logdump to its potential, please refer to the [Logdump Reference for Oracle GoldenGate](#) documentation.

Enable Debug Logging for Oracle GoldenGate Microservices

You can enable debug logging in the Oracle GoldenGate Microservices framework, which enables you to diagnose all the processes happening in a particular service. It can be enabled for any of the HTML5 development services. The following steps help you to enable debugging for the services:

1. Log in to Distribution Service or any service HTML5 pages.
2. Open the context menu to display options for that page.
3. Select Debug Log.
4. Click the `Enable Debug Log` toggle switch, to enable it.