Oracle® GoldenGate

Using Oracle GoldenGate Studio

26ai
G49527-01
February 2026

Oracle GoldenGate Using Oracle GoldenGate Studio, 26ai

G49527-01

Contents

4 Manage

5 Monitor

6 Troubleshoot

Preface

This preface describes the document accessibility features and conventions that are used in *Oracle GoldenGate Studio Documentation*.

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

Oracle GoldenGate Documentation

Oracle GoldenGate for Distributed Applications and Analytics

Oracle GoldenGate Studio Documentation

OCI GoldenGate

Oracle Database High Availability

Oracle GoldenGate Veridata

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1

Start

Get started with GoldenGate Studio by exploring new features and installing the product.

GoldenGate Studio Release Notes

Review release information before you install and work with GoldenGate Studio. This document is accurate at the time of publication and is updated periodically with each new software release.

What's New in GoldenGate Studio

Discover new features in GoldenGate Studio 26ai.

- GoldenGate Studio 26ai supports creating connections with MySQL variants.
- GoldenGate Studio for MySQL supports One-way replication recipe.
- New Database Migration recipe added with Oracle and MySQL.

GoldenGate Studio 23.9 New Features

Discover new features in GoldenGate Studio 23.9.

The new features in GoldenGate Studio 23.9 are:

- Deployment management, where you can manually register and manage multiple replication environments for better scalability and control.
- Support for Oracle Autonomous Database Wallets for secure, encrypted credential management, meeting enterprise compliance and security needs.
- Flexible Initial Load options with different transfer mediums, including Database Link, Object Storage, and File Storage.
- A new replication recipe, ZeroETL Mirror, adding a powerful solution for enterprise scale data integration projects.

What's supported

GoldenGate Studio 26ai supports new variants of Oracle AI Database and MySQL Databases as source and target connections. See Supported Connections to know more.

Known Issues

Learn about known issues within the current release version.

Release 26ai - January 2026

Bug 38365169: Generic - Minimum shared pool memory setting needed to avoid job failing at later stages

Oracle GoldenGate doesn't allow updating or increasing the shared pool memory on runtime.

Workaround

To avoid this shared memory exhaustion, you can perform one of the following solutions:

- Allocate certain fixed percentage of total database size as shared memory pool.
- Decrease the parallelism while data is being processed in shared memory pool. However, this can reduce the performance of Extract.

Bug 38745592: Test connection fails for Oracle 19 Container Database connection

Test connection to Oracle 19c Container Database (CDB) fails with an error, while the same connection details function successfully with other database clients.

Workaround

Use a database user created by GoldenGate.

Bug 38848546: Clicking on save icon always required two times in Mapping Configuration page

The Save icon on the mapping page requires two clicks to update the target name; the first click has no effect.

Workaround

None

Bug 38750565: Scroll not working in Pipeline Configuration Mapping tab

Occasionally, scrolling in the **Pipeline Configuration Mapping** tab is unresponsive.

Workaround

Refresh the browser.

Bug 38855648: Oracle-to-Oracle Database Migration: Wait time for open transactions is not handled in offline migration

The configured wait time for open transactions is not applied during offline migrations; the pipeline continues without stopping when the wait time elapses.

Workaround

Ignore the **Wait Time for Open Transactions** option while using the offline migration use case.

Bug 38900081 - Pipeline configuration Initial Load options Data pump label needs modification

The Initial Load Data Pump label appears in MySQL migration and replication pipelines, reflecting Oracle-specific behavior. For MySQL, existing target tables are not skipped and only data differences are loaded. This is a cosmetic issue and does not affect the functionality.

Workaround

None

Bug 38833864 - Pipeline remain in ACCEPTED state during Oracle-to-Oracle Offline/ Online Database Migration or One-Way Replication

When running online or offline database migration and One-way replication between Oracle OnPrem databases, the pipeline job remains in the ACCEPTED state and does not progress. This issue occurs in the initial run and resolves on subsequent re-runs.

Workaround

Re-run the pipeline.

Bug 38898013 - Intermittent - No stop button on CDC involved pipeline when is is successfully running.

Intermittently the stop button may not appear on CDC involved pipeline monitoring page. It corrects the state on reload.

Workaround

Reload the page.

Bug 38875701 - Pipeline main page not reflecting correct status for pipelines

The main pipeline page incorrectly displays the status as Not Started for all pipelines except those marked as Failed, even after successful completion.

Workaround

Click on **Reload** in browser.

Bug 38905451 - Offline Migration (non Autonomous AI Database to Autonomous AI Database): Pipeline do not fail if Private Network is not selected for Database Link

During offline migration from non Autonomous AI Database to Autonomous AI Database, the pipeline remains in the Accepted state and does not fail when **Database on Private Network** option is not selected with Database Link as the Initial Load transfer medium.

Workaround

Always select the **Database on Private Network** checkbox for this use case.

Bug 38908821 - In Pipeline monitoring, failed job keeps reloading API polling, which should have stopped once the job is failed

API polling and GoldenGate Studio interface refresh continues for failed jobs in pipeline monitoring, rather than ending after job failure.

Workaround

Ignore the GoldenGate Studio interface flashing due to polling.

Bug 38910713 - End user could not navigate to Running Jobs Detail page

Sometime selecting any running pipeline on the Runtime tab makes GoldenGate Studio interface unresponsive and prevents navigation to other pages, specially when many pipelines are running simultaneously.

Workaround

Refresh the browser.

Release 23.9 - October 2025

Check the known issues in GoldenGate Studio 23.9.

Bug 38493323: Active-Active - Pipeline failed at Prepare database objects for extraction on target

Pipeline fails while preparing database objects for extraction on target for schema/table mapping.

Workaround

Oracle recommends that you create the user from the GoldenGate Studio application itself, instead of assigning an externally created user.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 38477069: Active-Active - Default conflict resolution type needs to be shown in the GoldenGate Studio user interface

The default conflict resolution type is not displayed in the GoldenGate Studio user interface.

Workaround

None.

Bug 38475961 - Delete all deployments doesn't work when we select them using the select all button

When all the deployments are selected together using the check box left to the headings, the Delete option doesn't work.

Fixed an issue where using the **Select All** option in deployments now allows **Delete All** to function as expected.

Workaround

Select each deployment individually and delete all at once.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 38461500 - Mapping tab in the user interface doesn't reflect the custom Mapping Rules selected in Mapping Rules tab

The mapping rules are not reflected in the Mapping page of the GoldenGate Studio user interface.

Workaround

None.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 38446349 - The status of some steps is not updated in the user interface even though API reflects it properly

Status of certain steps is incorrectly displayed in the user interface while the API reflects the correct status.

Fixed an issue where step statuses in the UI are now updated to match the API

Workaround

None.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 38442543 - Certain special characters are not supported in SCHEMA or table name

If special characters are used in SCHEMA or table name then you will not able to edit or save the pipeline.

Workaround

Use asterisk (*) instead of using special characters when saving the mapping.

Bug 38425814 - Exclude disappears from Mapping rules if Include *.* is deleted

If the **Include *.*** option is deleted from the Mapping Rules table in the Pipeline Configuration Mapping Rules tab, the user interface deletes the **Exclude** option instead of the **Include *.*** option from the user interface.

Workaround

Delete ***.*** mapping first to avoid this issue.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 38415742 - Active-Active: Could not create pipeline with connection having period in the service name

If an Active-Active pipeline is created with a period in the service name, the pipeline creation fails.

Workaround

None.

Bug 38389703 - When re-running a pipeline, the Extract and Replicat processes are shown stopped though they are actually running in Oracle GoldenGate deployment

After a pipeline is restarted, the Extract and Replicat processes are shown in stopped state although they are running in the Oracle GoldenGate deployment.

Workaround

None.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 38384257 - Even after stopping a job , we see stop job option

Stop button is enabled in the GoldenGate Studio user interface even though the job is initializing and the Extract/Replicat processes are not running.

Workaround

None.

Bug 38314733 - User interface inconsistent with backend status while monitoring Pipelines

Pipeline page in the user interface gets stuck sometimes.

Workaround

None.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 38285621 - GoldenGate Studio pipeline configuration page is stuck when source or target Oracle Autonomous AI Database is not reachable

When there are a lot of connections to an Oracle Autonomous AI Database along with multiple pipelines, GoldenGate Studio keeps on polling the jobs status APIs even when they aren't responding. This creates slowness and the system may get stuck.

Workaround

Ensure that the Oracle Autonomous AI Database is running so that the pipeline configuration page shows valid data.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Bug 35927051 - Incorrect Mapping Rules for a specific mapping pattern

Mapping rules are not saved correctly when a specific mapping pattern is selected in the Mapping Rules page.

Workaround

None.

> **Note**
>
> This issue is fixed in the 26ai (23.26.1.0.0) release.

Schema Mapping and Configuration

When you finish creating a pipeline by clicking on the "Save & Configure" button, you are shown the Mappings section, where all the source's schemas are listed and they all come preselected. If you use accessibility keys such as tab, shift, return, then you can't select or deselect any of the mapping options, nor display all schemas/tables for the drop down list.

Workaround

Use the mouse instead of using the accessibility keys.

Database Version Mismatch

The Initial Load step may fail if the source database version is higher than the target database version causing errors similar to the following:

- **ORA-39059:** Dump file set is incomplete.
- **ORA-39246:** Cannot locate master table within provided dump files.

Workaround

Before initiating replication, ensure that the target database is upgraded to at least the same version as the source database.

How to Get Help

Use My Oracle Support to find knowledge solutions, workaround, and other information that is reported by customers, partners, and Oracle employees. My Oracle Support also enables you to open a Service Request. If a patch is required to resolve a service request, you will receive instructions on how to download it from 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.

> **Note**
>
> If you purchased Oracle GoldenGate Studio Commercial and support through a distributor, contact your distributor instead of attempting to create a service request through My Oracle Support.

Use the Community Forum if you are using GoldenGate Studio Free.

Licensing Information

Visit Oracle GoldenGate Licensing Information to review detailed licensing compliance information for Oracle GoldenGate Studio.

Install GoldenGate Studio

The GoldenGate Studio installation process involves preparing your environment, downloading the appropriate software package, and following a guided setup to ensure that all necessary components are correctly configured.

Before You Begin

To use Oracle GoldenGate Studio, ensure that you have:

- A system running Linux
- A valid Oracle.com login
- JDK 25 installed and configured on your system

GoldenGate Studio Software Update

Apply the latest software patch to your GoldenGate Studio installation to ensure you have the most recent features and fixes. Follow these steps to patch your current GoldenGate Studio Software.

If you already have GoldenGate Studio 23.9.0 installed, proceed to apply the latest patch (23.26.x). See Apply Latest Software Release Patch.

If Studio 23.9.0 is not installed, continue with Get the Installer section.

Get the Installer

To download the GoldenGate Studio installer:

1. Open the Oracle Software Delivery Cloud portal: https://edelivery.oracle.com/.
2. Click **Sign In**, and then enter your Oracle account username and password.
3. After you log in, enter **GoldenGate Studio** in the search bar.
4. Click the **Add to Cart** button corresponding to the GoldenGate Studio version that you want to download.

> **Note**
>
> Download the most recent version available to ensure you receive the latest features and security updates.

5. Unzip the Shiphome. Use an appropriate tool (`unzip`, for example) to extract the archive.
6. Navigate into the extracted `disk` directory to access the setup files.

Run the Installer

To install GoldenGate Studio:

1. Set the `JAVA_HOME` environment variable to point to the supported JDK version.
2. Open the command prompt and navigate to the location where the installer files are located.
3. Navigate to the `Disk1` folder and run the following command to start the Installation Wizard:

```
cd Disk1
./runInstaller -jreloc $JAVA_HOME
```

4. On the **Installation Type** screen, the Oracle GoldenGate Studio option is preselected. Click **Next**.
5. On the **Specify Installation Details** screen, browse and select the location where you want to install GoldenGate Studio, and then click **Next**. For example, `/u01/app/ggstudio/`
6. On the **Create Inventory** screen, you can choose a location to store the installation metadata files such as the install.log. Also select the user group that is allowed to access this inventory location. By default, the **dba** user group is selected. Click **Next**.
7. On the **Summary** screen, review the settings, and then click **Install**. You can also save the reponse file for later reference using the **Save Response File** option.

Set Up Deployments Using Oracle GoldenGate Studio Configuration Assistant Wizard

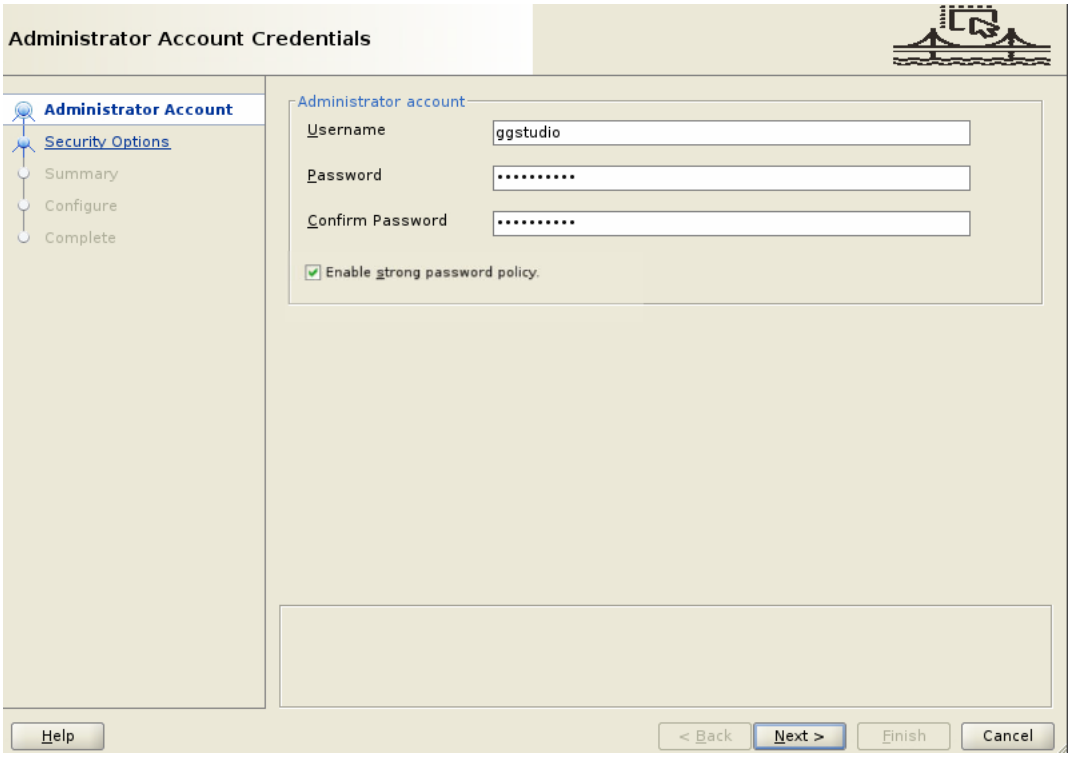
Run the GoldenGate Studio Configuration Assistant wizard to set up and access GoldenGate Studio deployments. Perform the following steps to start the wizard and configure the deployment credentials and security:

> **Note**
>
> You must have Java version 25 or higher installed on your system.

1. Update the `PATH` environment variable to use JDK 25:

   ```
   export PATH=$JAVA_HOME/bin:$PATH
   ```

   The `PATH` environment variable is required to run Java commands such as `java`, `javac` using the supported JDK package.
2. From the GoldenGate Studio installation home directory, navigate to the `bin` directory. For example: `cd /u01/app/oggstudio/bin`
3. Run `./oggstudioca.sh`. This opens the Oracle GoldenGate Studio Configuration Assistant wizard.
4. On the **Administrator Account Credentials** screen, specify the **Username** and **Password** for your deployment user account, as shown in the following image.



Use these credentials to log in to Oracle GoldenGate Studio for the first time

The strong password policy is enabled by default. It's recommended that you use a strong password for your administrator account. The criteria for a strong password includes:

- 1 uppercase letter (A - Z)
- 1 lowercase letter (a -z)
- 1 digit (0 - 9)
- 1 special character (- ! @ % & * . # _)
- Password length must be between 8 and 30 characters

If you need to change the credentials after the deployment setup completes, you can run the `setPassword.sh` script to modify the credentials of the Administrator Account user. This script is located in the `bin` directory of the GoldenGate Studio home directory (`$STUDIO_HOME`). Here are the steps to run this script:

a. When you run the script, it asks for the existing user name of the Administrator Account. Specify the user name and press enter.

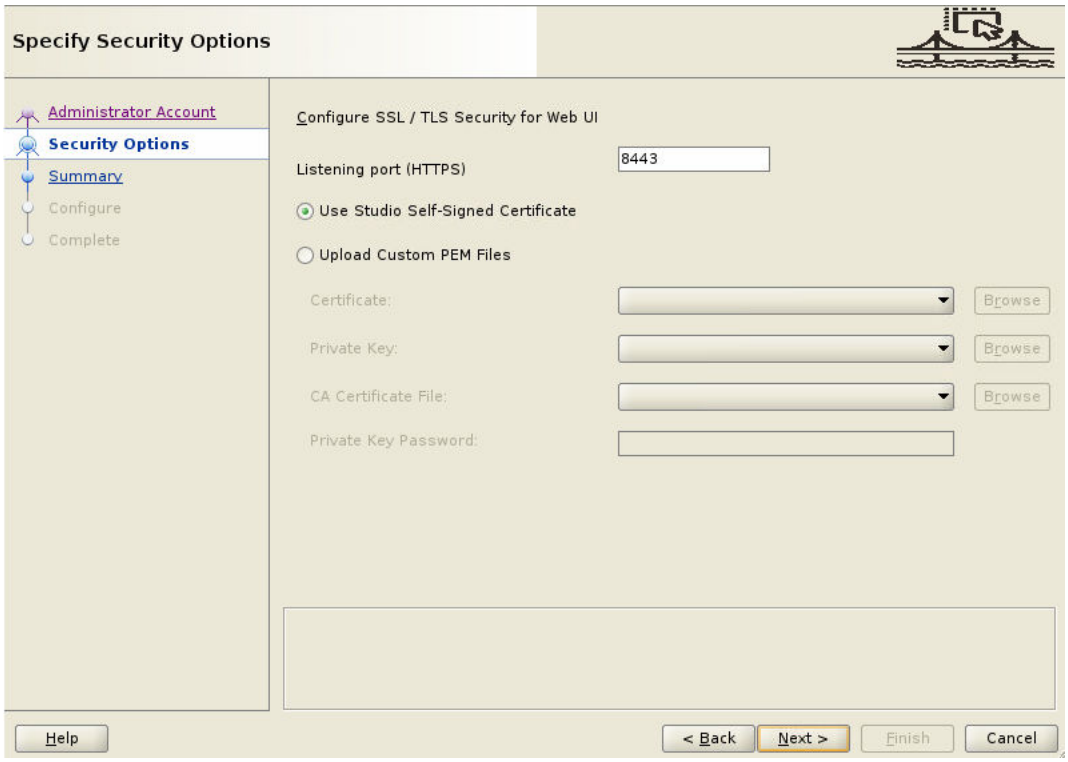
b. Now, specify the new password for the specified user. The scripts prompts you to confirm the password. After the new password matches, the system displays the following message:

```
The studio.conf file updated successfully.
```

Now, you will be able to use the new password to log in to the GoldenGate Studio user interface.

5. On the **Specify Security Options** screen, specify a **Listening Port Number** on which GoldenGate Studio will run.

> **Note**
>
> By default, the **Listening Port** for the deployment is 8443. You can use different port also.



6. GoldenGate Studio allows you to use self-signed certificates or `.pem` files for authentication.
If you choose **Self-Signed Certificates**, then GoldenGate Studio automatically generates a self-signed SSL certificate. This option is useful in a non-production environment where a trusted CA certificate is not required.

   If you choose **Upload Custom PEM Files**, complete the following fields:

a. **Certificate**: This is the public certificate with a `.pem` extension that will be used to identify the server to clients. Browse and select the certificate file.

b. **Private Key**: Browse and select the private key for the certificate.

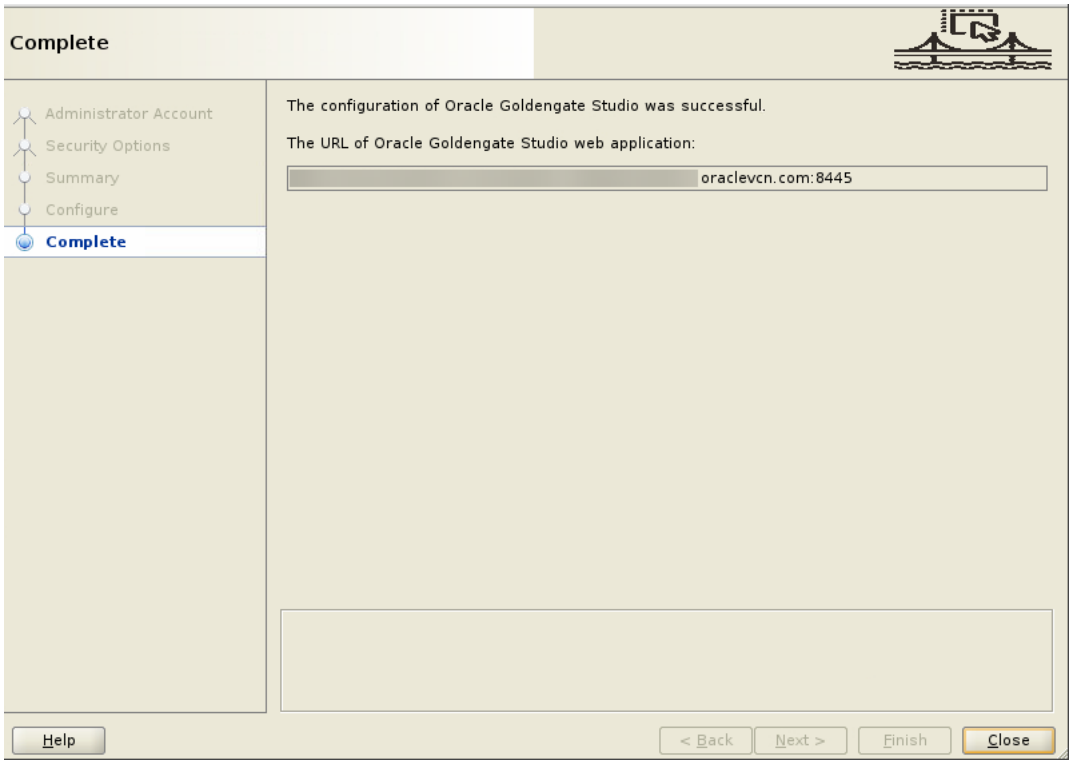
c. **CA Certificate File**: This file contains the certificate(s) of the Certificate Authority (CA) that signed the server/client certificate. The CA file is used to verify that the server certificate is trusted.

d. **Private Key Password**: Enter the password for the private key that must be available with the private key file.

This option is best suited for production environments where an official SSL certificate issued by a trusted CA.

7. Click **Next**.

8. On the **Summary** screen, review the values that you provided in the previous screens and save the response file to a location of your choice. Click **Finish** to begin the configuration process.
The Configure screen shows the progress of the configuration process. After the process completes successfully, it displays the Complete screen, where you can copy the web URL of the GoldenGate Studio.



Apply Latest Software Release Patch

Follow these steps to apply the latest software release.

Use this procedure to apply the latest Oracle GoldenGate Studio patch release (e.g., 23.26.1.0.0) to your existing installation.$STUDIO_HOME is the "Software Location" specified during Oracle GoldenGate Studio installation (see step 5 in Run the Installer).

1. Download the latest patch from Oracle support site, e.g. 23.26.1.0.0 or higher.
2. Unzip the patch file to create a folder named with the patch number (for example, 38821818); this folder is referred to as the patch folder.
3. Set `JAVA_HOME` to `JDK 25`.
4. Stop the GoldenGate Studio Server.

   Command:

   ```
   cd $STUDIO_HOME/oggf/bin
   ./run-studio.sh stop
   ```

5. Export `ORACLE_HOME` variable to your GoldenGate Studio installation location.

   Command:

   ```
   export ORACLE_HOME=$STUDIO_HOME
   ```

6. Navigate into patch folder.

   Command:

   ```
   cd <into patch folder in step 2>
   ```

7. Apply patch to the GoldenGate Studio installation.

   Command:

   ```
   $ORACLE_HOME/OPatch/opatch apply
   ```

8. Check if the patch is installed properly.

   Command:

   ```
   $ORACLE_HOME/OPatch/opatch lsinventory
   ```

Connect to the GoldenGate Studio User Interface

Connect to the GoldenGate Studio user interface after the GoldenGate Configuration Assistant Wizard completes to verify your installation.

Use the URL generated by the GoldenGate Studio Configuration Assistant Wizard to access the GoldenGate Studio user interface.

To open GoldenGate Studio:

1. In your web browser, enter the deployment URL generated by the GoldenGate Studio Configuration Assistant Wizard. The GoldenGate Studio login page is displayed.
2. Enter the Administrater Account login credentials you created while setting up the deployment in theGoldenGate Studio Configuration Assistant Wizard. If your log in is successful, you're brought to the GoldenGate Studio Home page.
3. On the GoldenGate Studio Home page, you can learn more about the product using the help options, or register a deployment to get started.

Stop/Restart the GoldenGate Studio Server

You can stop and restart the GoldenGate Studio server using the `runstudio.sh` script, located in the `$STUDIO_HOME/bin` directory.

To stop GoldenGate Studio server instance, run the following command:

```
./run-studio.sh stop
```

To restart the GoldenGate Studio server instance, run the command:

```
./run-studio.sh
```

Install GoldenGate Studio Free

Discover what you need and how to get started with GoldenGate Studio Free.

What You Need

To use GoldenGate Studio Free, you need the following:

- A system running one of the following:
  - Docker, or a compatible container runtime such as Podman
  - Linux
- Internet connectivity to access the container registry
- A valid Oracle.com login

From Oracle Container Registry

Follow these instructions to access the GoldenGate Studio Free from Oracle Container Registry (OCR).

- You must sign in to your Oracle account and accept the license agreement before you can pull the GoldenGate Studio Free docker image from the Oracle Container Registry.
- Obtain an Auth token from the Oracle Container Registry page and use this as your password along with your username.

To access the GoldenGate Studio Free docker image:

1. Accept the license agreement.
2. In your Docker client, enter the following command to log in to the Oracle Container Registry:

   ```
   docker login container-registry.oracle.com
   ```

3. When prompted, enter your Oracle account username and the Auth token password.

4. Copy the `docker pull` command to pull the latest GoldenGate Studio Free image, and run it in your Docker client:

```
docker pull container-registry.oracle.com/goldengate/goldengate-studio-
free:latest
```

Start and run the GoldenGate Studio Free docker image

Learn to start and run GoldenGate Studio Free after pulling the docker image from Oracle Container Registry.

To start and run GoldenGate Studio Free:

1. In your Docker client, you can enter a run command with default settings or enter a run command with parameter values:
   - To run GoldenGate Studio Free with default settings:

     ```
     docker run -p 8443:8443 -e STUDIO_ADMIN=studioadmin -e
     STUDIO_ADMIN_PWD=<password> container-registry.oracle.com/goldengate/
     goldengate-studio-free:latest
     ```

     > **Note**
     >
     > Ensure that you replace `<password>` with a password of your choice.

     A container instantiates with the following default options:
     - An auto generated container name
     - 80 port for HTTP Server
     - 443 port for HTTPS Server
     - `studioadmin` for the name of the administrative account
     - An auto generated self-signed certificate
   - To run GoldenGate Studio Free with modified settings:

     ```
     docker run \
         --name <container name> \
         --hostname <domain name> \
         -p <host port>:8443 \
         -e STUDIO_ADMIN=<admin user name> \
         -e STUDIO_ADMIN_PWD=<admin password> \
         container-registry.oracle.com/goldengate/goldengate-studio-
     free:latest
     ```

     Parameter descriptions:
     - `--name <container-name>`: Name for the container. Auto-generated by default.
     - `--hostname <domain name>` - The domain name to use for subject name and DNS in the self-signed certificate
     - `-p <host-port>:8443`: The host port to map the Oracle GoldenGate HTTPS server. No mapping by default.

- `-e STUDIO_ADMIN`: The name of the administrative account to create. by default.
- `-e STUDIO_ADMIN_PWD`: The password of the administrative account.

> **Tip**
>
> Ensure the password meets the following requirements:
>
> * At least one lowercase character [a...z]
> * At least one upposercase character [A...Z]
> * At least one digit [0...9]
> * At least one special character [- ! @ % & * . #]
> * The length should be between 8 and 30 characters.

2. After a few minutes, you should see `Server started at: https://<host-address>:8443`. GoldenGate Studio Free is now ready for you to use.
3. Launch a web browser, and go to `https://<host-address>:8443`.
4. On the GoldenGate Studio Free login screen, enter the username and password that you passed to the docker run command, and then click **Log in**.

After you log in successfully, you're brought to the GoldenGate Studio Free Home page.

Logging

Oracle GoldenGate Studio maintains logs for server and performance analysis, while also allowing debugging of issues by viewing drill-down details in the log.

You can also view the initial load logs for different source and target database combinations.

Server Logs

Explore how to set up and access server and performance logs.

- You can find error message logs for GoldenGate Studio server and performance in the `log` directory. This directory is located in `$STUDIO_INSTALLER_DIR/oggf/log`.
- To search for a GoldenGate Studio server log message, you need to search for the server log name using the name **`ggos-api-server%g.log`**.
  To search for a performance log message, use the name **`ggos-api-perf%g.log`**.

  The **`%g`** represents index of the file used to maintain the log rotation.
- You can also access logs in the GoldenGate Studio user interface. To view details about the cause and action of an error message, select the error code (OGGOS-60000, for example).

You can configure different levels of logging for GoldenGate Studio Server and performance logs. For more information, see Configure Log Levels for Server and Performance Logs.

Initial Load Logs

There are two types of logs generated for the initial load process:

- `EXPDAT*.log`: Export log from the source database.

- `IMPDAT*.log`: Import log from the target database.

The location of the log files depends on the transfer medium (DBLink, Local Storage, and Object Storage) and type of database (Autonomous Database, or non-Autonomous Database). To learn about log retrieval of these log files, see Retrieve the Initial Load Log Files.

Uninstall GoldenGate Studio

Before you begin the uninstallation process, make sure that you stop any GoldenGate Studio instances.

Perform the following steps to uninstall GoldenGate Studio:

1. Shut down a running GoldenGate Studio instance by executing the following script:

   ```
   ./STUDIO_HOME/bin/run-studio.sh stop
   ```

2. Set the `JAVA_HOME` environment variable to point to a supported JDK version. Ensure that Java version 24 or higher is installed on your system.
3. Run the following script to uninstall:

   ```
   ./STUDIO_HOME/deinstall/deinstall.sh -jreLoc $JAVA_HOME
   ```

2

Introduction

Learn about GoldenGate Studio concepts to help you get started.

About GoldenGate Studio

GoldenGate Studio helps you to design and deploy high-volume, real-time replication by automatically handling table and column mappings and generating best practice configurations from recipes. With GoldenGate Studio, you can start replicating data using a simple interface. GoldenGate Studio supports features such as multiple deployment management, authentication using Oracle Wallet for secure credential management forOracle Autonomous AI Database, and validation utilities like **Test Connection** for reliable environment setup.

GoldenGate Studio is available in two editions: a **free version** with community-based support, and a **commercial version**. The commercial is ideal for organizations that require comprehensive Oracle Support, including access to regular upgrades and patches, ensuring a stable and maintainable environment for production use. This edition is particularly beneficial for organizations running mission-critical workloads, as it offers the ongoing support and regular updates necessary to keep complex environments secure and up to date. By providing full Oracle Support, the commercial version ensures smooth operations and minimizes risk for enterprise data integration initiatives.

> **Note**
>
> The **Database Migration recipe** is now available. For MySQL, only **One-way Database replication** and **Database Migration** recipes are supported. Oracle AI Database supports **One-way Database replication**, **Database Migration**, **Active-Active Database Replication** and **ZeroETL Mirror** recipe. Check Supported Oracle Databases and Supported MySQL Databases .

GoldenGate Studio delivers an enterprise-ready platform with advanced features, broader compatibility—including support for newly added Oracle and MySQL databases—scalable deployment options, and access to Oracle support.

GoldenGate Studio provides you with the following abilities:

- Quickly create, modify, validate, deploy, and reuse Oracle GoldenGate best practice designs.
- Define data flows once and deploy to numerous locations.
- Graphically filter, map, and transform data.
- Apply global rules and exceptions.

Who is GoldenGate Studio for?

GoldenGate Studio is ideal for database administrators, architects, and IT professionals responsible for setting up and maintaining data integration and replication solutions.

GoldenGate Studio is ideal for a wide range of users involved in data integration and replication projects, especially database administrators, data architects, and IT teams responsible for doing replication across multiple Oracle databases and locations. For those already using GoldenGate, GoldenGate Studio will enhance the experience by offering a highly intuitive, visual interface that significantly simplifies the process of designing, deploying, and managing replication environments. Its guided workflows make it easy to configure and monitor even complex data integration setups, reducing manual effort and streamlining the entire lifecycle of GoldenGate projects.

GoldenGate Studio makes it easier for both new and experienced GoldenGate users to handle everything from simple data integrations to complex, enterprise-wide deployments.

What can I use GoldenGate Studiofor?

GoldenGate Studio enables you to visually design, deploy, and monitor end-to-end data replication and integration environments with GoldenGate.

GoldenGate Studio makes it simple and efficient to set up, deploy, and manage real-time data replication solutions across Oracle AI Database. It helps you generate and validate GoldenGate configurations based on best practices—allowing you to quickly build, modify, and reuse replication projects without writing manual scripts. With GoldenGate Studio, you can define powerful data flow designs and deploy them simultaneously to multiple locations, ensuring that data stays synchronized across all your critical systems.

Features like filtering, mapping, and transforming data are easily configured in GoldenGate Studio. This helps you align replication with your specific business logic. Organizations can ensure that only the right data is replicated, in the right format, to meet their operational and analytical needs. This means data replication isn’t just a copy-paste activity; it’s tailored to support your specific use cases, compliance needs, or reporting standards.

GoldenGate Studio Concepts

Get familiar with the following concepts and other commonly used terms before you get started with GoldenGate Studio.

Concept	Definition
Deployment	The process of setting up and activating your data replication design so it can run in a real environment. A GoldenGate Studio deployment is GoldenGate's instance, configured to run pipelines. Learn how to Register Deployments.
Connection	Contains the connectivity information for source and target databases used in pipelines. Learn how to: • Create Oracle Database and Oracle Exadata Connection • Create Oracle Autonomous Database Connection
Recipe	A predefined template in GoldenGate Studio that automates the design and creation of data integration solutions, specifying how source and target systems interact. Learn about Recipes.
Pipeline	A pipeline enables replication across diverse database environments. Learn how to Create Pipelines.
Mapping Rules	Rules for including or excluding schemas and tables in a data replication.
Advanced Options	Editable process or load parameters for Extract, Replicat, and data pump operations. Learn more: • Initial Load Advanced Options • Extract Advanced Options • Replicat Advanced Options

Concept	Definition
Extract	Process that reads or captures database changes from the source database.
Replicat	Process that applies captured changes to the target database.

GoldenGate Studio Limitations

Learn about the limitations of using GoldenGate Studio.

There are certain limitations and considerations that you should be aware of when using GoldenGate Studio.

- GoldenGate Studio doesn't currently support Oracle Cloud Infrastructure GoldenGate deployments.
- GoldenGate Studio works in top down approach. It generates GoldenGate Connections, Extracts, Replicats, and Trail Files based on connections and pipelines created in GoldenGate Studio.
- Existing Connections, Extracts, Replicats, Trail Files in GoldenGate can not be used by GoldenGate Studio pipelines.
- Refrain from changing Extract, Replicat, and parameter files that are generated by GoldenGate Studio. Once these files are changed, editing or running these pipelines from GoldenGate Studio lead to inconsistencies and failures.

3 Replicate

Follow the basic taskflow to replicate data in no time.

The Replicat process in GoldenGate Studio follows these steps, in sequence:

1. Register a deployment. A single GoldenGate deployment must be used for both source and target databases.
2. Create database connections. Specify the connection details (such as host, port, service name, username, and password) for the source or target database to enable data replication design and deployment.
3. Create pipelines. Use one of the included recipes to get started.
4. Configure pipelines. Add mapping rules and configure advanced options.
5. Monitor the pipelines. Check the real time visual pipeline diagram, that updates as you make changes to the pipeline configuration.

Deployments

GoldenGate Studio connects to an Oracle GoldenGate on-premise instance and mirrors the replication processes from the GoldenGate Studio user interface, allowing further processing.

About Deployments

A GoldenGate Studio deployment process maps a designed and tested data replication solution from Oracle GoldenGate.

A GoldenGate Studio deployment generates and moves the necessary configuration and parameter files to the GoldenGate Studio target servers, so that the Oracle GoldenGate processes can be started and managed in a production environment.

With GoldenGate Studio deployments, you can:

- Connect and manage multiple Oralce GoldenGate environments from a single interface, providing flexibility and centralized control. Rather than designing pipeline for every environment, you can just reuse your existing pipeline and change its target to deploy to a different environment.
- Use different GoldenGate deployments for testing, development, or production. This means you can safely build and test your pipelines before running them in your live environment.
- Update, modify, or relocate them directly on the target systems without impacting your design work or project configurations within GoldenGate Studio.

  This means if you make changes directly to Oracle GoldenGate configurations or processes on those servers (for example, editing parameter files, changing process settings, or moving processes to another server), those changes do not automatically affect or update the design and settings stored in GoldenGate Studio.

- Manage each deployment's lifecycle separately, performing operations like start, stop, upgrade, or delete without affecting other environments. For example, while upgrading the test deployment, the production deployment continues running without interruption.
- Create new deployments to support additional teams, projects, or regions. For example, expanding into a new business unit or region simply requires spinning up a new deployment to handle its unique replication needs.

  You must register at least one deployment before you create and run any pipelines.

Register Deployments

Learn to register Oracle GoldenGate deployments in GoldenGate Studio.

To register a deployment:

1. From the navigation menu, click **Deployments** and then click **Register Deployment**.
2. In the **Register Deployment** panel, complete the fields as follows:
   a. Enter a **name** of the deployment.
   b. (Optional) Enter a **Description**, to help distinguish this deployment from others.
   c. Enter deployment's **Hostname** or IP address.
   d. Enter the **Port** number of the deployment's Administration Service.
   e. Enter the **Username** of the Oracle GoldenGate administrator user. See Authorization in Oracle GoldenGate .
   f. Enter the **Password** of the Oracle GoldenGate administrator user.
   g. Select **Use Reverse Proxy** *only* if it's configured for the Oracle GoldenGate instance.
3. Click **Test Connection**. If the deployment connects successfully, then the **Next** button becomes activate.
4. Click **Next**. On the Confirmation screen, review the deployment details, and then, click **Save**.

Database Connections

Database connections store the credentials and parameters required to access databases that serve as sources or targets in GoldenGate Studio. This configuration enables secure and efficient communication for designing and deploying data replication pipelines.

Database connections in GoldenGate Studio represent the configuration information required to communicate with your source and target databases. These connections are the foundation for designing, deploying, and managing data integration and replication solutions within the GoldenGate Studio. They enable you to visually map the data flow across diverse environments.

For details, see the Database Connection, System and Parameter Settings in *Oracle GoldenGate Microservices Documentation*.

About Database Connections

Learn about the connection types you can use with GoldenGate Studio and how to create them.

A database connection refers to the configuration information and credentials GoldenGate Studio needs to connect to a database. This connection allows GoldenGate Studio to access and interact with the source and target databases as part of designing, deploying, and managing data replication solutions.

GoldenGate Studio currently supports Oracle AI Database connections.

Supported Connections

Check the following list for supported Oracle Database variants and MySQL databases you can use as sources and targets in GoldenGate Studio.

GoldenGate Studio supports Oracle Database versions 19*c*, 21*c*, 21*c* XE, 23ai, 23ai Free, and Oracle AI Database 26ai. Starting with GoldenGate Studio 26ai, additional variants of Oracle database are supported. See Supported Oracle Databases.

GoldenGate Studio 26ai and higher support source and target connections to MySQL database variants. See Supported MySQL Databases for the complete list.

Supported Cloud Platforms for Oracle and MySQL Database

Learn about the supported cloud platforms for Oracle and MySQL databases.

When planning cloud deployments of Oracle and MySQL databases, it is important to consider the range of supported cloud platforms. Both database technologies can be deployed across various leading cloud service providers, including:

- Oracle Cloud Infrastructure (OCI)
- Amazon Web Services (AWS)
- Micosoft Azure
- Google Cloud Platform (GCP)

Supported Oracle Databases

GoldenGate Studio supports Oracle AI Database platforms, including on-premises, cloud, and multi-cloud environments.

The following table lists all currently supported Oracle AI Database types and deployment options, along with key connection requirements and supported recipe types.

> **Note**
>
> You can use Oracle AI Database for older versions of Oracle Databases.

Technology Type	Deployment Environment	Supported Recipes
Oracle AI Database	On-premises	All
Oracle Exadata Database	On-premises	All
Oracle Autonomous AI Database	Oracle Cloud	All
Oracle Database@Azure - Autonomous Database	Microsoft Azure	All
Oracle Database@Azure - Exadata	Microsoft Azure	All

Technology Type	Deployment Environment	Supported Recipes
Oracle Database@AWS - Autonomous Database	Amazon Web Services	All
Oracle Database@AWS - Exadata	Amazon Web Services	All
Oracle Database@Google Cloud - Autonomous Database	Google Cloud Platform	All
Oracle Database@Google Cloud - Exadata	Google Cloud Platform	All
Amazon RDS for Oracle	Amazon Web Services	All

Supported MySQL Databases

GoldenGate Studio enables you to configure integration and migration pipelines for various MySQL databases, including their leading managed cloud offerings.

The following table outlines all supported MySQL databases, the available recipe types, and important configuration details:

Technology Type	Deployment Environment	Supported Recipes
MySQL Database Server	On-premises	One-way, Database Migration
Amazon Aurora MySQL	AWS Cloud	One-way, Database Migration
Amazon RDS for MySQL	AWS Cloud	One-way, Database Migration
Azure Database for MySQL	Microsoft Azure Cloud	One-way, Database Migration
Google Cloud SQL for MySQL	Google Cloud Platform	One-way, Database Migration
HeatWave MySQL on OCI	Oracle Cloud	One-way, Database Migration
HeatWave MySQL on Azure	Microsoft Azure	One-way, Database Migration
HeatWave MySQL on AWS	Amazon Web Services	One-way, Database Migration

Best Practices for Connections

Discover best practices for Connections in GoldenGate Studio.

Plan Before Creating the Connection

- **Identify Connection Requirements:** Gather details such as database type, hostname, port, service name/SID, user credentials, and SSL requirements before starting.
- **Verify Access Privileges:** Ensure the user account has the required permissions for GoldenGate operations (e.g., replication privileges, read/write access to specific schemas).

Use Secure Credentials Management

- **Avoid Hardcoding Credentials:** Store usernames and passwords securely in the GoldenGate credential store instead of saving them in plain text. Do not store passwords in scripts or plain text.
- **Oracle Wallet or Oracle Cloud Wallet services:** Use Oracle Wallet or Oracle Cloud Wallet services to store Oracle Autonomous Database credentials and encryption keys securely.
- **Use Role-Based Accounts:** Create separate database accounts for replication activities, with only the necessary privileges.

- **Change Default Passwords:** Immediately update default or shared account passwords to meet your organization's security policies. Regularly rotate passwords according to your organization's security policy.

Ensure Compatibility

- **Verify Database Version:** Check that your source and target databases are compatible with the GoldenGate version you are using.
- **Install Required Drivers:** Ensure JDBC drivers for the database type are installed and configured in GoldenGate Studio.

Optimize Connection Settings

- **Enable SSL/TLS if Available:** Use encrypted connections to protect data in transit.
- **Set Connection Timeout:** Configure timeouts to prevent long waits in case of connectivity issues.
- **Use Connection Pooling Where Possible:** This improves performance for multiple Oracle GoldenGate processes using the same connection.

Validate the Connection

- **Test the Connection in Studio:** Use the **Test Connection** feature in GoldenGate Studio to confirm connectivity and authentication.
- **Verify Schema Access:** Ensure the connected user can access the specific schemas, tables, and views needed for replication.
- **Check Performance:** Run small test queries to assess response time before deploying the connection in production.

Maintain and Monitor

- **Document the Connection Details:** Maintain an internal record of connection parameters, privileges, and owner for auditing.
- **Regularly Rotate Credentials:** Update stored passwords periodically to comply with security policies.
- **Monitor Connection Health:** Use Oracle GoldenGate monitoring tools to check for connection drops, slow queries, or authentication failures.

Create Oracle AI Database and Oracle Exadata Connections

Learn how to create Oracle AI Database and Oracle Exadata connections.

1. From the GoldenGate Studio Home page, click **Create connection.**
   You can also create a connection from the **Connections** page.
2. The Create Connection panel consists of three pages. On the **General Information** page, complete the following fields, and then click **Next**:
   a. Enter a **Connection Name.**
   b. (Optional) Enter a **Description**.
   c. Select a **Technology Type** from the dropdown menu.
3. On the Connection Details page, complete the following fields, and then click **Next**:
   a. Enter the database **Username** and **Password**.

b. Enter the database **Hostname** and **Port**

c. For **Database Type**:

- If your Oracle AI Database version is 21c or above, select **Pluggable database (PDB) in Oracle Database 21c or above**.
- If your Oracle AI Database version is 19c or below, select **Pluggable database (PDB) in Oracle Database 19c or below** and provide both the **Pluggable database service name** and the **Container database service name**
- If your Oracle AI Database version is Non-container database, select **Non-container database** and then enter the **Database service name**.

4. On the Confirmation page, click **Create**.

Configure and Download SSL Wallet for Non Autonomous AI Database

Learn about the different **Wallet Configuration** options.

Wallet configuration is required in non Autonomous AI Database to enable secure connectivity. This setup facilitates verified communication with endpoints such as parUrl using SSL verification or other Oracle Autonomous AI Database hosted in secure regions over HTTPS, especially needed to upload and download files from secured `parUrl` / `bucketUri` in initial load using Object Storage.

Why Is Wallet Configuration Important?

Without a properly configured wallet, attempts to connect securely to Object Storage or other endpoints using HTTPS will fail, leading to errors or inability to upload/download files. Configuring a wallet reduces these risks by ensuring that your database trusts the cloud endpoints you're connecting to.

Important Terms

Term	Definition
parUrl	Pre-authenticated Request URL for secure, temporary access to objects in Oracle Cloud Object Storage.
bucketUri	A Uniform Resource Identifier pointing to a specific bucket in Oracle Cloud Object Storage. A bucket is a storage container for objects (such as files or data dumps).
Oracle Cloud Endpoint	URL for connecting to Oracle Cloud services.
ACL (Network Access Control List)	Security rules that define which network traffic is allowed or denied to network resources.

Standard Oracle Wallet

Learn about Standard Oracle Wallet.

An Oracle Wallet is a secure, encrypted container that stores authentication and encryption credentials like private keys, certificates, and trusted certificates. It ensures that sensitive data is not kept in clear-text configuration files. You create these wallets using tools like the Oracle Wallet Manager or `mkstore`, and they require a password to open and access contents. To enable secure connectivity from a non-autonomous database, wallet setup is required.

Follow these steps to download/configure Standard Oracle Wallet:

- Download the master wallet containing pre configured Oracle certificates. Contact the database administration team for this.
- Extract the certificate files to a directory on the database host file system.
- Specify this directory path in the SSL Wallet Path fields under the **Advanced Options**.

> **Note**
>
> - This wallet should serve the purpose for establishing secure connection with `parUrl` or `bucketUri` (typically used in case of Initial load via Object Storage).
> - Any required Network ACL configurations are automatically managed by the Studio application at runtime when access is initiated.

Using the Oracle-supplied wallet automatically enables secure connectivity for standard Oracle Cloud endpoints (like `parUrl` and `bucketUri`).

Manual Wallet Configuration

Learn how to configure Manual Wallets:

The **Manual Wallet** approach allows users to create and configure a wallet by manually adding specific security certificates that are not included in the Standard Oracle Wallet. This method is necessary when connecting to custom endpoints, using third-party certificate authorities, or addressing unique security requirements.

Follow the below steps to download/configure Manual Wallet:

- Download the required certificates for the `parUrl`- typically a root certificate and an intermediate certificate.
- Log in to the database host and create a wallet directory.
- Initialize a wallet using orapki utility available in Oracle base.
- Add certs downloaded from `parUrl` to wallet using `orapki` utility.
- Commands to be used are :
  - `bash-4.2$ mkdir -p <wallet-path-accessible-to-db-user>`
  - `bash-4.2$ orapki wallet create -wallet <wallet-path-accessible-to-db-user> -pwd <pwd> -auto_login`
  - `bash-4.2$ orapki wallet add -wallet <wallet-path-accessible-to-db-user> -trusted_cert -cert <path-to-root-certificate> -pwd <pwd>`
  - `bash-4.2$ orapki wallet add -wallet <wallet-path-accessible-to-db-user> -trusted_cert -cert <path-to-intermediate-certificate> -pwd <pwd>`
  - `bash-4.2$ orapki wallet display -wallet <wallet-path-accessible-to-db-user>`

Create an Oracle Autonomous AI Database Connection

Learn to create a connection to Oracle Autonomous AI Database to use as a GoldenGate Studio source or target.

Before you Begin

Before you create a connection, ensure that you:

- Turn on **Archive Log Mode** and restart your database.
- Download the Oracle Autonomous AI Database instance's wallet from the Oracle Cloud console. If you don't have access, contact your administrator.

To create an Oracle Autonomous AI Database connection:

1. From the **GoldenGate Studio** home page, click **Create connection**.
2. The Create Connection panel consists of three pages.
   a. On the **General Information** page, complete the fields as follows, then click **Next**:
      i. Enter a **Connection Name**.
      ii. (Optional) Enter a **Description**.
      iii. Select Autonomous AI Database, from the **Technology Type** dropdown.
   b. On the **Connection Details** page, complete the fields as follows, and then click **Next**:
      i. Enter a **Username** and **Password** for the GoldenGate admin user.
      ii. Click **Add Wallet File** and upload the Oracle Autonomous AI Database instance's wallet.
      iii. Click **Test Connection**.
   c. On the Confirmation page, verify the connection details.
3. Click **Create**.

Create MySQL Database Connection

Learn about the prerequisites for creating a MySQL database connection and the steps to set up the connection from GoldenGate Studio.

Before You Begin

Before running MySQL workloads that interact with Oracle GoldenGate or similar tools, ensure that the database environment satisfies all prerequisites.

To apply configuration changes, you need to edit the MySQL configuration file (`my.cnf`) and restart MySQL. If you do not have the required permissions to edit the `my.cnf` file, then contact the database administrator to attain the required privileges.

The following table includes the prerequisites, required action, and step-by-step guidance for implementation.

MySQL Prerequisite	Action Required	How to Set Up
Unsupported data types columns	Remove or convert columns with unsupported types.	1. Run the following query: `SELECT TABLE_SCHEMA, TABLE_NAME, COLUMN_NAME,COLUMN_TYPE FROM information_schema.COLUMNS WHERE COLUMN_TYPE IN ('geometry', 'linestring', 'polygon', 'multipoint', 'multistring', 'multipolygon', 'geometrycollection','set');` 2. `ALTER TABLE` to drop or change these columns (ask database administrator if needed)
Timezone match	Set Oracle GoldenGate server timezone to be the same as the MySQL server timezone.	1. Check the MySQL timezone: `SELECT @@system_time_zone;` 2. Check Oralce GoldenGate host timezone: `date` (ask database administrator if needed)
`ALLOW_INVALID_DATE` in `sql_mode`	Remove from `sql_mode` or avoid invalid date entries.	1. Run the following command: `SHOW VARIABLES LIKE 'sql_mode';` 2. Edit `my.cnf` and remove `ALLOW_INVALID_DATE` from `sql_mode`. 3. Restart MySQL.

AWS RDS log retention	Set `binlog_retention_hours` to at least 24.	1. Run the following command:
		`CALL mysql.rds_set_configuration ('binlog retention hours', 24);`
		2. Check the current configuration parameters and their values for your RDS instance:
		`CALL mysql.rds_show_configuration;`

`binlog_format`	Set `binlog_format = ROW`	1. Run the following command: `SHOW VARIABLES LIKE 'binlog_format';` 2. Edit `my.cnf`, set `binlog_format=ROW`. 3. Restart MySQL.

MySQL Prerequisite	Action Required	How to Set Up
`binlog_transaction_compression`	Set `binlog_transaction_compression = OFF`	1. Run the following command: `SHOW VARIABLES LIKE 'binlog_transaction_compression';` 2. Edit `my.cnf`, set `binlog_transaction_compression=OFF`. 3. Restart MySQL.
Character set	Use only supported character sets at all levels.	1. Run the following command: `SHOW VARIABLES LIKE 'character_set_server';` 2. If using unsupported value, then edit `my.cnf` and restart.
Collation server	Set `collation_server` to *`your_charset`*`_ci` (not binary)	1. Run the following command: `SHOW VARIABLES LIKE 'collation_server';` 2. Edit `my.cnf` and set `collation_server=`*`your_charset`*`_ci`. For example, `collation_server=utf8mb4_ci`.. 3. Restart MySQL.

Storage engine (capture/apply)	Use only InnoDB tables.	1. Run the following query:
		`SELECT TABLE_NAME, ENGINE FROM information_schema.TABLES WHERE TABLE_SCHEMA = '`*`your_schema`*`';`
		2. Run the following command:
		`ALTER TABLE `*`table`*` ENGINE=InnoDB;`

Database version	Upgrade to MySQL 5.7 or above.	1. Run the following command: `SHOW VARIABLES LIKE 'version';` 2. If you have lower version, then ask admin to upgrade.

MySQL Prerequisite	Action Required	How to Set Up

Database user privileges (capture)	Grant the following privileges:`SELECT, REPLICATION SLAVE, REPLICATION CLIENT, SHOW VIEW`	1. Run the following query:
		`SELECT Select_priv, Repl_slave_priv, Repl_client_priv, Show_view_priv FROM mysql.user WHERE User='ogg_user';`
		2. To grant user privileges:
		`GRANT SELECT, REPLICATION SLAVE, REPLICATION CLIENT, SHOW VIEW ON *.* TO 'user'@'%'; FLUSH PRIVILEGES;`

Database user privileges (apply)	Grant the following privileges or permissions:
	`SELECT, CREATE, CREATE VIEW, EVENT, INSERT, UPDATE, DELETE, DROP, EXECUTE`
		1. Run the following query:
		`SELECT Select_priv, Create_priv, Create_view_priv, Event_priv, Insert_priv, Update_priv, Delete_priv, Drop_priv, Execute_priv FROM mysql.user WHERE User='user';`
		2. To grant the privileges:
		`GRANT SELECT, CREATE, CREATE VIEW, EVENT, INSERT, UPDATE, DELETE, DROP, EXECUTE ON *.* TO 'user'@'%'; FLUSH PRIVILEGES;`

DDL replication only greater than and equal to 8.0	Upgrade to MySQL 8.0 plus and set `binlog_row_metadata = FULL`.	1. Run the following commands:
		`SHOW VARIABLES LIKE 'version';`
		`SHOW VARIABLES LIKE 'binlog_row_metadata';`
		2. Edit `my.cnf` for `binlog_row_metadata=FULL;`
		3. Restart MySQL.

Functional Indexes	Remove functional indexes.	1. Review the table definition, then run the following: `ALTER TABLE table DROP INDEX index_name;`

MySQL Prerequisite	Action Required	How to Set Up
Group replication	Enable `gtid_mode` or use special Oracle GoldenGate paramaters.	1. Run the following command: `SHOW VARIABLES LIKE 'gtid_mode';` 2. Set the `_DISABLEGTIDRECOVERY_true` parameter in Oracle GoldenGate.

Key columns (PK/UK)	Add a primary key or unique key to each table.	1. Run the following query:
		`SHOW INDEX FROM table WHERE Non_unique=0;`
		2. To add primary key:
		`ALTER TABLE table name ADD PRIMARY KEY (column); or ALTER TABLE table ADD UNIQUE (column);`

MariaDB binlog annotation	Set `binlog_annotate_row_events = OFF` for MariaDB 10.2.	1. Run the following command:
		`SHOW VARIABLES LIKE 'version';`
		2. Run the following command:
		`SHOW VARIABLES LIKE 'binlog_annotate_row_events';`
		3. Edit `my.cnf`, set `binlog_annotate_row_events=OFF`;
		4. Restart MariaDB.

No LOB datatype in key column	Avoid using `BLOB` or `TEXT` columns as primary key.	1. Run the following query: `SELECT COLUMN_NAME, DATA_TYPE FROM information_schema.COLUMNS WHERE COLUMN_KEY='PRI' AND (DATA_TYPE='blob' OR DATA_TYPE='text');` 2. Redesign the table, if needed.
No XA transactions	Avoid XA transactions on captured databases.	1. Check: `XA RECOVER`. 2. If needed, set `binlog-ignore-db` in `my.cnf` for non-captured databases. 3. Ask database adminstrator for application change.

MySQL Prerequisite	Action Required	How to Set Up
Server ID for remote capture	Set `server_id` to greater than 0.	1. Run the following query: `SHOW VARIABLES LIKE 'server_id';` 2. Set `GLOBAL server_id=1;` or set in `my.cnf` 3. Restart MySQL.
`STRICT_TRANS_TABLES` in `sql_mode`	Add `STRICT_TRANS_TABLES` to `sql_mode`.	1. Run the following query: `SHOW VARIABLES LIKE 'sql_mode';` 2. Edit `my.cnf`, add `STRICT_TRANS_TABLES`. 3. Restart MySQL.
Supported database versions	Use supported database versions only. Refer to the certification matrix for supported database versions.	1. Run the following query: `SHOW VARIABLES LIKE 'version';` 2. Check cloud/variant-specific queries.
Single JSON-column table	Do not use single-column tables with only a JSON column for replication.	1. Run the following query: `SELECT table_name FROM information_schema.tables WHERE table_schema='schema';` 2. Check your design and alter the table as needed.
`binlog_row_image`	Set `binlog_row_image = FULL`.	1. Run the following command: `SHOW VARIABLES LIKE 'binlog_row_image';` 2. Edit `my.cnf` and set `binlog_row_image=FULL`; 3. Restart MySQL.
`log-bin`	Set `log_bin = ON`.	1. Run the following command: `SHOW VARIABLES LIKE 'log_bin';` 2. Edit `my.cnf` and set `log_bin=ON`; 3. Restart MySQL.

Create the MySQL Database Connection

Follow these steps to create MySQL database connection:

1. From the GoldenGate Studio home page, click **Create connection**.

2. The Create Connection panel consists of three pages.
   a. On the General Information page, complete the fields as follows, and then click **Next**:
      - Enter a Connection Name.
      - (Optional) Enter a Description.
      - Select any MySQL Database, from the **Technology Type** dropdown.
3. On the Connection Details page, complete the fields as follows, and click **Next**:
   a. Enter a Username and Password for the GoldenGate admin user, Hostname, Port, and Database Name.
   b. From the Security Protocol dropdown menu, select one of the following options:
   - **Plain:** Select this option if you do not require encrypted communication. This sends information without any security.
   - **TLS:** Select this option if you require Transport Layer Security (TLS) to encrypt communications between your client and the server. If you select **TLS**, you will be prompted to specify how TLS is used. Select any of these options:
     - **Required:** It mandates an encrypted connection and terminates the session if TLS is not supported, ensuring maximum security.
     - **Preferred:** It attempts to use encrypted TLS connections first. If TLS negotiation fails, the connection automatically falls back to unencrypted mode for compatibility.
4. Click **Test Connection**.
5. On the Confirmation page, verify the connection details.
6. Click **Create**.

Pipelines and Recipes

Learn to create data replication processes using recipes and pipelines.

About Pipelines

A pipeline is an instance of a recipe. It enables you to select your source connection, the type of replication action(s) to apply, and the target connection. After a pipeline starts, you can observe the replication process in real time.

If you're familiar with Oracle GoldenGate Extract and Replicats, pipelines are an abstraction of the Oracle GoldenGate replication process. When you start a pipeline, Oracle GoldenGate:

- Prepares the database objects for extraction
- Creates a Heartbeat Table
- Creates and runs the Extract process
- Performs an initial load using Oracle Data Pump
- Creates the Checkpoint Table
- Creates and runs the Replicat process

These steps can be observed during the pipeline's Initialization phase on the Pipeline Details page.

You can view pipelines on the Home page, as well as the Pipelines page. Before you create a pipeline, ensure that you have Database Connections configured for the source and target databases.

To know more, see About Extracts and About Replicats in *Oracle GoldenGate Microservices Documentation*.

About Recipes

Recipes are templates for common replication scenarios that you can use to accelerate your data replication journey.

A recipe is a template that defines a set of data replication tasks or mappings between source and target systems. Recipes help simplify and accelerate the creation of replication solutions. Using a recipe reduces the time and effort required to set up new data integration projects, ensuring consistency and reliability across different environments. GoldenGate Studio includes the following recipes:

- **One-way Database Replication**: One-way data replication is a process in which data changes from a source database are continuously or periodically copied to a target database, but not vice versa. Only the source updates the target; changes in the target database do not flow back to the source. Performs an initial load and ensures the target database is kept in sync with the source database.
  Use one-way data replication in the following scenarios:
  - **Reporting/Analytics:** You want to offload queries and analytics to a separate reporting/BI database without affecting your primary transactional database.
  - **Data Migration:** Moving data from a legacy system to a modern one with as little downtime as possible.
  - **Disaster Recovery/Backup:** Maintain an up-to-date standby instance for failover/failback scenarios.
  - **Data Distribution:** Send data from headquarters to branch office databases.
- **Active-Active Database Replication**: **Active-Active replication** is a data architecture in which **two or more databases** can accept read and write operations simultaneously, and changes are automatically synchronized between all sites in near real time. This enables continuous availability and seamless failover and supports distributed workloads. Performs an initial load of the source database to the target, and then applies change events in both directions to keep the databases synchronized.

  Use Active-Active replication in the following scenarios:
  - When seamless disaster recovery and high availability are required, allowing all sites to process traffic and support immediate failover.
  - When balancing workloads across multiple, geographically dispersed data centers to improve overall system performance and resilience.
- **Database Migration:** Provides a step-by-step framework for transferring data from one system or environment to another using Oracle tools and best practices. It typically involves planning, data assessment, extraction, transformation, loading, validation, and post-migration support. The Database Migration recipe performs an Initial load, and then applies change events until you're ready to switch over to the target database.

> **Note**
>
> GoldenGate Studio does not support data migration from Oracle AI Database (including Oracle Autonomous AI Database) to MySQL database.

Database migration is necessary when you want to perform a one-time or phased bulk movement of database objects and data from one environment to another (for example, during cloud adoption or hardware refresh). The data migration recipe facilitates online and offline database migrations:

- **Offline:** Source applications must be taken offline to prevent updates to the source database during migration. Offline migration is most suitable for small databases or where the duration of downtime is not a concern. Data and metadata are exported from the source database and imported into the target database. Offline migration is ideal for development and testing in small, non-critical database environments.

  > **Note**
  >
  > For MySQL, only the One-way recipe and Data Migration recipe are supported.

- **Online:** Source systems can remain online during the migration. Online migration is most suitable for large databases or for critical applications where the duration of downtime is a concern. An initial snapshot of data and metadata is exported from the source database and imported into the target database, followed by continuous synchronization of changes.

- **ZeroETL Mirror:** ZeroETL is a data integration approach that eliminates traditional, batch-oriented ETL pipelines. Provides near real-time replication from an operational database to an analytics or reporting environment without requiring traditional ETL (Extract, Transform, Load) processes. This recipe simplifies architecture and delivers up-to-date data for analytics with minimal latency.
  ZeroETL is ideal for scenarios where actionable insights from fresh operational data are crucial—for example, real-time dashboards, instant reporting, integrating legacy on-premises databases with other enterprise applications, and seamless cloud or on-premises migrations.

  > **Note**
  >
  > If you encounter a warning about sequences detected, ensure that you don't replicate database generated sequential values. The range of values must be different on each system, with no chance of overlap. For example, in a two-database environment, you can have one server generate even values, and the other odd. For an *n*-server environment, start each key at a different value and increment the values by the number of servers in the environment. This method may not be available to all types of applications or databases. If the application permits, you can add a location identifier to the value to enforce uniqueness.

Create a Pipeline

Learn to create pipelines in GoldenGate Studio.

Before You Begin

Before you create a pipeline, ensure that you've created and tested the source and target database connections. For Oracle AI Database, database configuration is performed during the **Configure Source** or **Configure Target** step while creating pipeline. For MySQL, you must configure the database before creating a pipeline. Check prerequisites for MySQL in **Before you Begin** section of Create MySQL Database Connection.

If you want to use Oracle Autonomous AI Database as your pipeline's source or target connection, you must:

1. **Configure Oracle Autonomous AI Database**:
   a. **Download the Wallet**: Download the Oracle Autonomous AI Database wallet from the instance's details page in Oracle Cloud Infrastructure. You'll upload it to GoldenGate Studio when you create the connection, if you haven't done so already.
   b. **Configure TNS Alias**: Use the alias from `tnsnames.ora` inside the wallet for the Oracle Autonomous AI Database connections.
   c. **Check Client Credentials**: Make sure username and password are available and have permission for the GoldenGate replication operations.
   d. **Verify Autonomous Database Firewall and VNet rules**: Ensure appropriate network access so GoldenGate Studio can reach your Oracle Autonomous AI Database.

   > **Note**
   >
   > Ensure that your wallet and credentials management comply with the Oracle security policies.

2. **Target Database Access**: Configure and validate connectivity to your target database (which can be Oracle Autonomous AI Database or any other supported Database).
3. **User Alias Configuration**:
   a. **Create a credential store with a user alias.** Complete steps 3 to 6 in *Configure Extract for Oracle Autonomous Database*. The user alias references the secure credentials for the source and target databases.
   b. Register the user alias in the GoldenGate credential store, and enter this user alias in GoldenGate Studio when creating pipelines with Oracle Autonomous AI Database connections to avoid storing plain-text credentials.

   > **Note**
   >
   > To verify if the user alias is functioning correctly, go to your **GoldenGate** user interface and test the Database Connections there first. See Prerequisites for Capturing from Autonomous Database in *Oracle GoldenGate Microservices Documentation*.

4. **Database Privileges**: You need the required database permissions on the source and target side to support GoldenGate operations (typically `SELECT`, `FLASHBACK`, DBA-level privileges, and so on).

> **Note**
>
> For MySQL migration and replication recipe to work, you must have permission to grant access to the necessary databases and tables. Execute the following SQL command to grant all privileges:
>
> ```
> GRANT ALL PRIVILEGES ON *.* TO 'username'@'host' WITH GRANT OPTION;
> FLUSH PRIVILEGES;
> ```

Create a Pipeline

To create a pipeline:

1. From the **GoldenGate Studio** Home page, click **Create Pipeline**.

   You can also select **Pipelines** from the navigation menu, and then click **Create Pipeline** on the Pipelines page.

2. The Create Pipeline panel consists of 6 pages:

   a. On the Recipe page, select a **Recipe**, and then click **Next**.

   b. On the Overview page, enter a **Name** and optionally, a **Description**, and then click **Next**.

   c. On the Connections page, select the **Source** and **Target Connections** and **Deployments**. If either of the connections is an Oracle Autonomous AI Database instance, you must also provide the **GoldenGate user alias**. Click **Next**.

   GoldenGate User Alias Requirements for Autonomous Database

Scenario	Required Alias (es)
Oracle Autonomous AI Database as Source only	Source GoldenGate user alias
Oracle Autonomous AI Database as Target only	Target GoldenGate user alias
Oracle Autonomous AI Database as Source and Target	Source and Target GoldenGate user aliases

   a. On the Validate Source page, click **Validate** to check and prepare the source database connection for replication. After the validation for source database completes successfully, click **Next**.

      If the source validation fails, then a new step, **Configure Source** is added. See Validate Source and Target Database Connections with SYS User to perform validation in this steps.

   b. On the Validate Target page, click **Validate**.

      If the target validation fails, then a new step, **Configure Target** is added. See Validate Source and Target Database Connections with SYS User to perform validation in this steps.

   c. On the Confirmation page, click **Save & Continue**.

3. You can now Configure the pipeline, or click **Start**.

After the pipeline starts, you can view detailed status and logs for each pipeline. This helps identify and troubleshoot errors, in case the pipeline creation process fails.

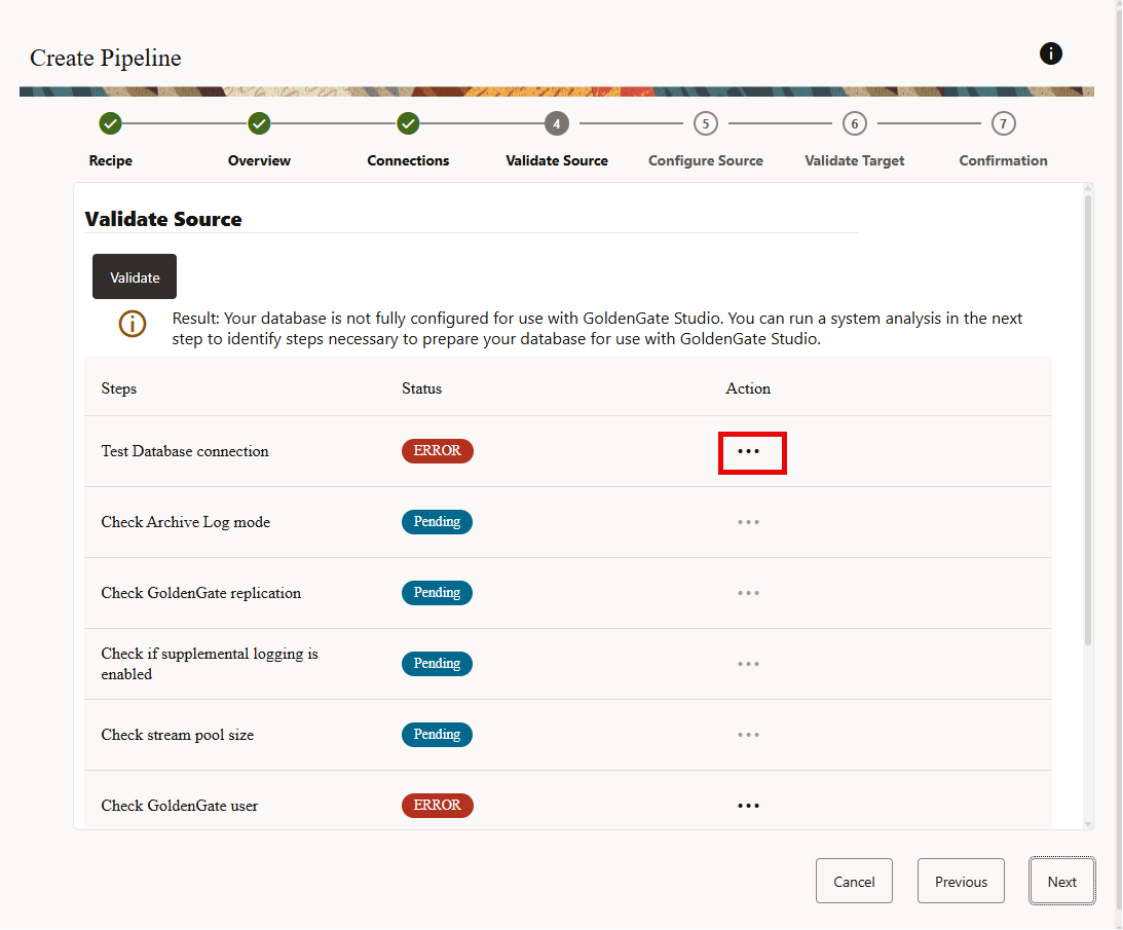
Validate Source and Target Database Connections with SYS User

This step is required only when the source and/or target database connection validation fails while creating a pipeline.

If a source validation step fails, then the validation source status displays an error.

> **Note**
>
> GoldenGate Studio only supports Configure option for Oracle database sources and targets in this release. For other technologies you need to manually configure your database.

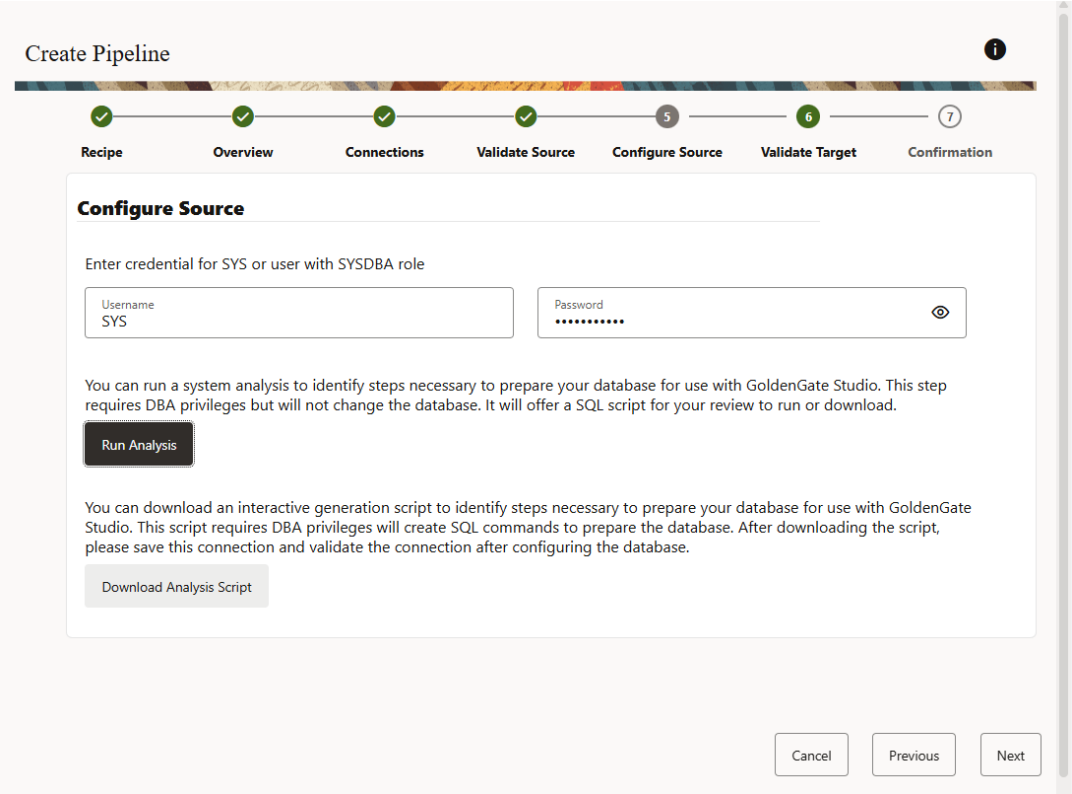
You can use the Action menu (ellipses) to view the error details for the validation failure, as shown in the following image:



The following steps are required to validate the source database connection using a SYS/SYSDBA user.

1. When the validate source connection step fails, click **Next** in the **Validate Source** screen. The **Configure Source** screen is displayed.

2. In the **Configure Source** screen, enter credentials for SYS user or a user with the SYSDBA role, as shown in the following image:



3. Select one of the following options:

> **Note**
>
> For Oracle CDBs, the SQL preparatory script requires connections to both the CDB (Container Database) and PDB (Pluggable Databases). The script automatically configures the necessary settings for both, the CDB and PDBs. The script should be run at the CDB level. If any ALTER DATABASE command requires a database restart, then the script will send a notification. To run this script, you require SYS user privileges.

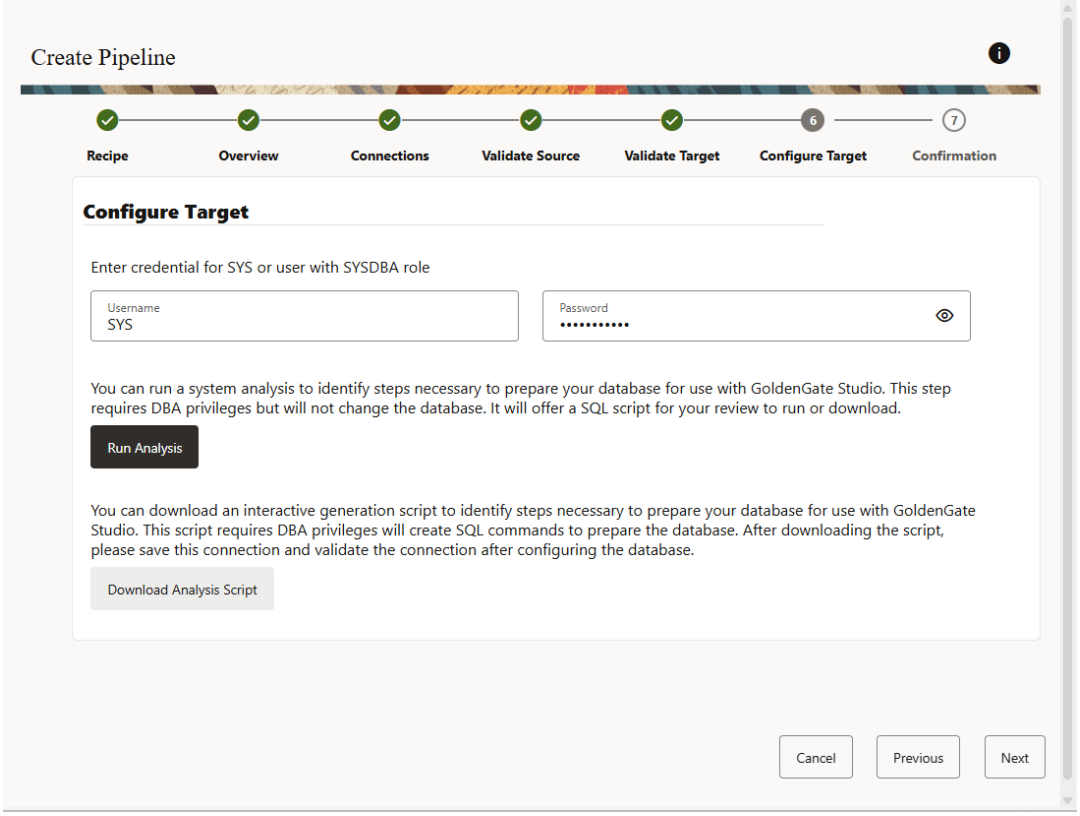
- Click **Run Analysis** to perform a system analysis and review the recommended SQL script needed to prepare your database for GoldenGate Studio (requires database administrator privileges).

  This option is available when the database is already configured and does not require additional configuration allowing you to execute the required scripts directly from the interface, streamlining the setup process.

- Click **Download SQL** to obtain a script that generates the required SQL commands for configuration.

  This option is available for databases that require manual configuration of enabling archive log, setting up `stream_pool_size`, or configuring GoldenGate replication for Oracle Database 19c, as these tasks may require a database restart. In these instances, you must first configure the database before proceeding.

4. If the target validation fails, then click **Next**. The **Configure Target** screen is displayed, as shown in the following image:



5. Follow the tasks given in step 3 for validating the target database connection using the SYS/SYSDBA user.

Configure a Pipeline

You must have a pipeline created before you can configure it.

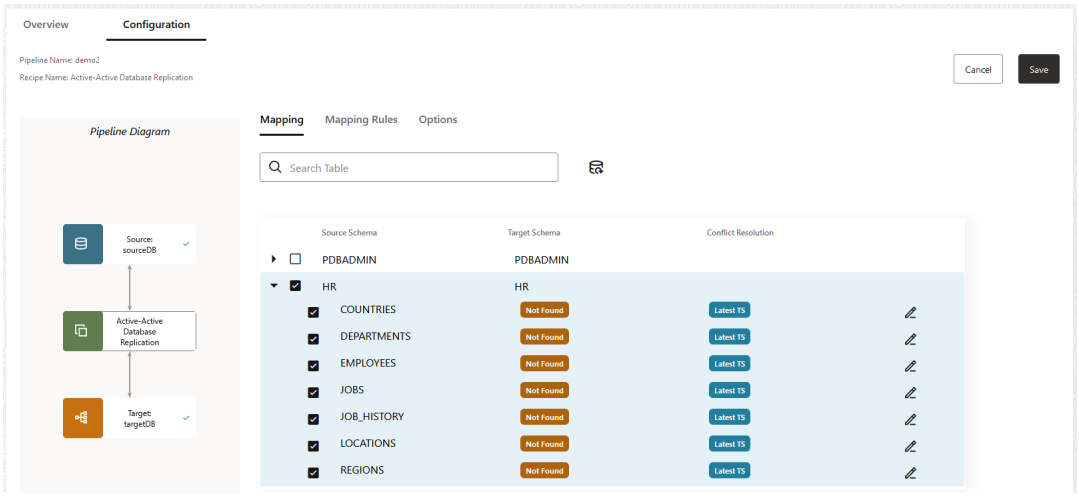
After you create a pipeline, the **Configuration** page is displayed. If you're revisiting a pipeline after creating it, you can select the pipeline from the **Pipelines** page to view its details, and then click **Configuration** to return to the Configuration page.

To configure a pipeline:

1. Under **Mapping**, you can review and select the source database schemas and tables to replicate.

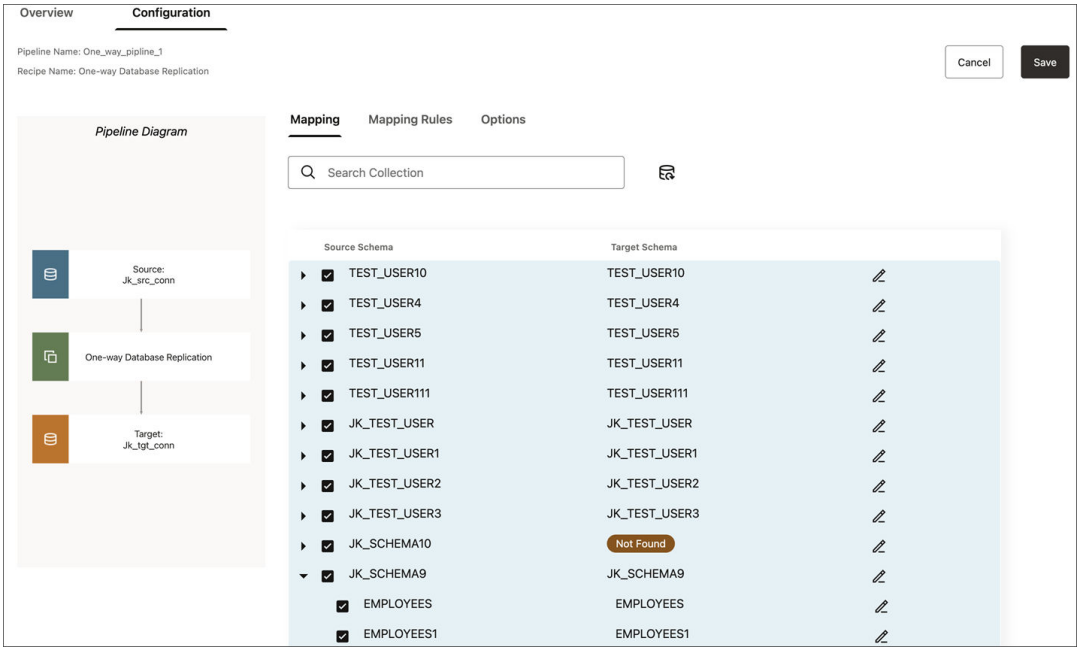
> **Note**
>
> The username entered when creating the connection will not appear on the pipeline **Mapping** page for mapping selection.

Always double check the rules added in the Mapping Rules list after selecting or deselecting schemas or tables using the **Mapping** tree view. These are the source of truth on what will be executed by Oracle GoldenGate.



> **Note**
>
> In One-way database replication, target schemas and tables show only when they exist in the target, otherwise they show as **Not found**.

The following image shows a One-way data replication pipeline:



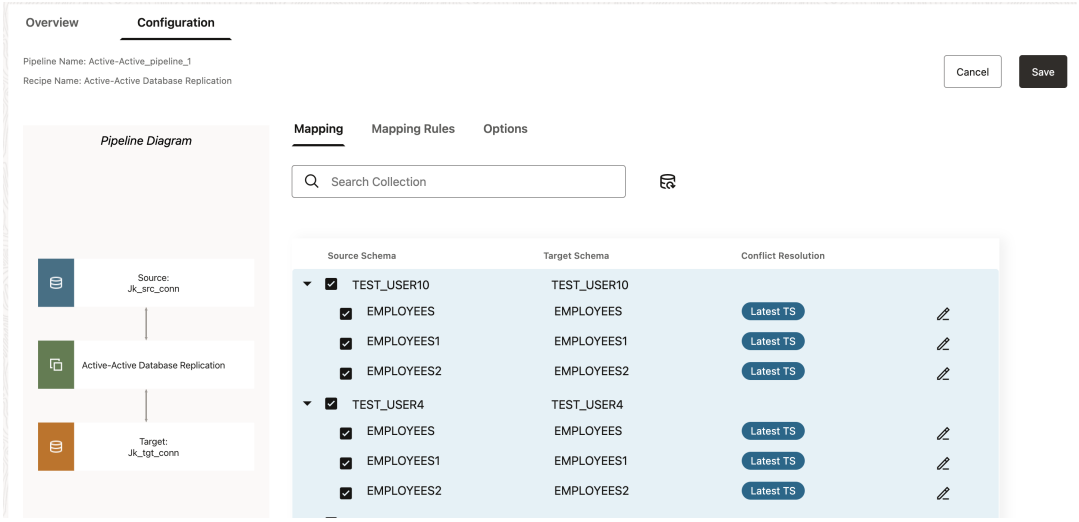
2. Using the **Mapping** tab, you can:
   - Click **Edit** (pencil icon) to rename a target schema or table. This redirects the replication to the renamed schemas and tables.
   - Include a schema and all its tables. This also ensures that any tables added to the schema in the future are also implicitly included and replicated.

- Include a schema but not all its tables. Tables **not** included will not be replicated, however, future tables added to the schema are implicitly included and replicated to the target.
- Exclude the schema but include its tables. Any tables added to the schema in the future are **not** included.
- Exclude an entire schema and all its tables.

As you select or deselect source schemas and tables, or edit target schemas or tables, rules are added to **Mapping Rules**.

> **Note**
>
> For Autonomous AI Database, system schemas such as DCAT_ADMIN, RMAN$CATALOG, GGADMIN, ADBSNMP, and ADB_APP_STORE may be visible. Consider excluding them manually from mapping.

3. For Active-Active replications, a **Conflict Resolution** column appears in the Mapping screen.



In Active-Active pipeline, Automatic Conflict Detection and Resolution (ACDR) is enabled for tables if it is not already enabled. If ACDR is already enabled, then GoldenGate Studio will not modify the configuration.

To configure Automatic Conflict Detection and Resolution (ACDR):

a. Click **Edit** next to the schema for whom you want to configure ACDR.

b. In the Edit table mapping panel, select **Automatic Conflict Detection and Resolution**.

c. For Timestamp, select whether the **Latest** change or **Earliest** change takes precedence.

d. Select **Delta Resolution** if the changes made must be combined to resolve the conflict, and then select the columns to use. For example, updates made to product inventory.

e. Click **Apply**.

Warning

GoldenGate Studio doesn't support the addition of new tables, nor the change of ACDR type, after the Active-Active pipeline starts. Unless absolutely necessary, you can either:

- Create, configure, and start a new pipeline with the new tables added.
- In the GoldenGate Studio console:
  a. Stop the pipeline.
  b. Create a new table externally.
  c. Under **Mapping**, expand the schema and select new tables.
  d. Under **Options**, in **Advanced Options** and under **Replicat**, select **DISCARD** for **Actions upon DML Error**.
  e. Save the pipeline configuration and then restart the pipeline.

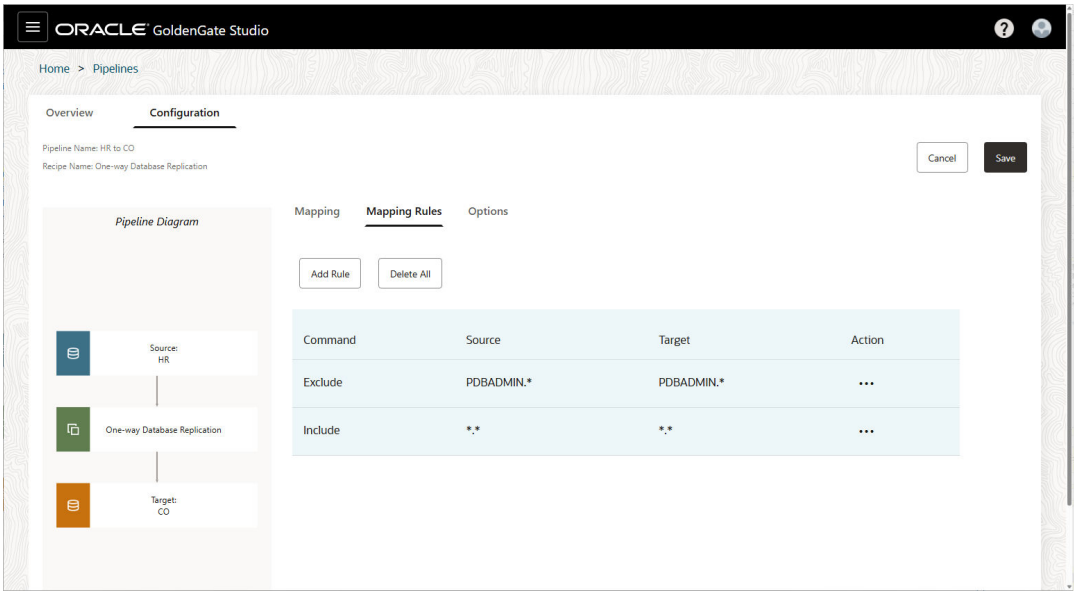
Note

ACDR isn't automatically enabled for tables without keys.

Note

If you stop a pipeline while ACDR is enabled or if the Enable ACDR step generates an error, then ACDR will not be enabled for selected tables.

4. Under **Mapping Rules**, you can review, add, delete, and reorder rules.



Additional Considerations:

- Exclude rules take precedence over Include rules.
- To delete a rule, click **Actions** (ellipsis icon) and select **Delete**.

- To reorder a rule, click **Actions** (ellipsis icon) and then select either **Increase priority** or **Decrease priority**.
- To add a new rule, click **Add rule**. You can use the following special characters when constructing rules:
  - Asterisk (*) as a wildcard for any number of characters
  - Underscore (_)
  - Space ( )
  - Double quotes (") to enclose schema or table names that include spaces or when case sensitivity is required.
- Individual rules take precedence over group rules.
- **Delete all** removes all rules in the list.

> **Note**
>
> Starting with Studio 23.26.1.X.Y.Z, adding a *.* mapping from the Mapping Rules tab translates to individual schema mappings. New schemas added after pipeline creation must be mapped manually from the **Mapping** or **Mapping Rules** page.

5. Under **Options**, you can configure a limited set of GoldenGate parameters:
   - **Initial Load of existing schemas and/or tables**

   > **Note**
   >
   > - If you use a **Database Link** for Initial Load, you must provide the Source Wallet URI parameter value.
   > - If you use **Object Storage** for Initial Load, you must provide the Object Storage Bucket URI value.

   - **Replicate Data Definition Language (DDL)**
   - **Advanced options** for Initial Load (Data Pump), Extract, or Replicat.

   > **Note**
   >
   > Refrain from making changes to the underlying parameter file as it affects the ability to manage the pipeline. Likewise, if using the GoldenGate Studio console, refrain from changing the Replicat type.

6. Click **Save** to save your configuration settings.
7. After configuring the pipeline, click **Start** to run it.

   You can review the initialization procedure, the status and progress of each step in the replication process, and how much time it takes for each step to complete.
8. After initialization completes, the **Runtime** page displays data capture operations on the source database replicated to the target.

The pipeline runs continually until it's stopped.

Next Steps

After your pipeline is created, configured, and running, learn to Manage Pipelines.

Configure Advanced Settings

Explore different advanced settings for GoldenGate Studio.

You can use **Advanced Options** to fine tune your replications. These settings allow you to optimize performance, customize deployment behavior, and manage environment-specific requirements.

Initial Load Advanced Options

When setting up a replication, you can configure advanced options to optimize the **Initial Load** process.

The Initial Load advanced options determine how the data is transferred, handled, and synchronized between the source and target systems. When configuring Initial Load options in Oracle GoldenGate Studio, available settings depend on the selected database type. This section outlines the available options for Oracle and MySQL databases.

Oracle AI Database: Supported Initial Load Advanced Options

Setting	Description	Value
Action Upon Existing Tables	Determines how to handle existing target tables during initial load.	`REPLACE, TRUNCATE, APPEND, SKIP`
Degree of Parallelism	Number of parallel threads for the Data Pump load. Higher values can speed up loads but increase resource usage. If the number of CPUs is 'n' and degree of parallelism value is less than or equal to 'n' then the degree value is going to be 'n'. However, if the number of CPUs are 'n' but the degree of parallelism value is 'x', which is greater than 'n', then the degree value is going to be 'x'	Any positive integer value >= 1
Additional Initial Load (Data Pump) Job Duration	Specifies for how much time jobs will run after the expected completion time for Initial Load. After the assigned time it will automatically time out.	`1h`
Transfer Medium	Specifies the transfer method for Initial Load.	`Database Link, Object Storage, File`
Object Storage Bucket URI	Location of the staging bucket for Initial Load files.	`https://objectstorage.us-phoenix 1.oraclecloud.com/`
Source Wallet URI	URI or path to the SSL Wallet for source database (required).	`/u02/app/oracle/admin/sourcedb/ssl_wallet`

Setting	Description	Value
Target Database SSL Wallet Path	This is Wallet directory location of SSL Wallet for target database.	`/u02/app/oracle/admin/targetdb/wallet` See Configure and Download SSL Wallet for Non-Autonomous Database
Source Database SSL Wallet Path	This is Wallet directory location of SSL Wallet for source database.	`/u02/app/oracle/admin/sourcedb/ssl_wal` See Configure and Download SSL Wallet for Non-Autonomous Database
Wait Time for Open Transactions	Duration to wait for open transactions to finish before replication starts.	`1h`
Export Directory (Local Shared Storage)	This is Source shared directory location for writing dump files during data pump export.	`/mnt/source_exports`
Import Directory (Local Shared Storage)	This is target shared directory location for writing dump files during data pump export.	`/mnt/target_imports`
Actions Upon Expiry	Specifies if there is need to continue or stop if the transactions are open.	`CONTINUE, STOP`

MySQL Database: Supported Initial Load Advanced Options

Setting	Description	Value
Action Upon Existing Tables	Determines how to handle existing target tables during initial load.	`SKIP, ERROR`

Setting	Description	Value
Degree of Parallelism	Number of parallel threads for the Data Pump load. Higher values can speed up loads but increase resource usage. If the number of CPUs is **'n'** and degree of parallelism value is less than or equal to **'n'** then the degree value is going to be **'n'**. However, if the number of CPUs are **'n'** but the degree of parallelism value is **'x'**, which is greater than **'n'**, then the degree value is	Any positive integer value >= 1.

Setting	Description	Value
	going to be **'x'**	

Setting	Description	Value
Transfer Medium	Specifies the transfer method for Initial Load.	`Database Link`, `Object Storage`, `File Storage`

Note

Database link is supported only for Oracle AI D

Setting	Description	Value
		database but not for MySQL Database.
Object Storage Bucket URI	Location of the staging bucket for Initial Load files.	`https://objectstorage.us-phoenix-1.oraclecloud.com/...`

Setting	Description	Value
Initial Load Dump Directory	Store dump files generated during Initial Load process. Dump directories are created under the ORACLE_BASE location for Oracle, and under the Studio installation directory for MySQL.	`db_dumps` (by default)

Note

This option applies when

Setting	Description	Value
		enFileStorageisselectedasTransferMedium.
Compression	Reduces the size of dump files	`NONE`, `GZIP`, `ZSTD`

> **Note**
>
> To get the Wallet Bucket URL from Oracle Cloud Infrastructure (OCI), where Oracle Autonomous AI Database is running, create a bucket in OCI and generate a pre-authenticated read/write URL for that bucket.

Data Compression Options

Learn different data compression options GoldenGate Studio offers.

The following data compression options are available in **Configuration Advanced Settings** for MySQL:

1. **NONE**

   Applies no compression. Selecting this option keeps files at their original size, enabling the fastest processing speeds but resulting in higher storage and bandwidth usage.

2. **GNU Zip (GZIP)**

   GZIP is one of the most established and widely used compression algorithms. Provides moderate compression rates and is widely supported across platforms. Selecting this option results in reduced file sizes and ensures compatibility with most tools and systems but may offer slower processing speeds compared to ZSTD. Use GZIP when you need maximum compatibility across tools and platforms, minimal CPU usage, or are working with smaller datasets where compression speed is not the main concern.

3. **Zstandard (ZSTD)**

   Zstandard (ZSTD) is a modern compression algorithm designed for both high compression ratios and rapid processing speeds. Offers higher compression efficiency and faster compression/decompression speeds than GZIP. Prefer ZSTD for large datasets, modern data pipelines, or performance-sensitive applications. It may not be supported by all legacy or third-party systems.

Type of Initial Load Transfer Mediums

In GoldenGate Studio, the **Initial Load** process is used to transfer a complete dataset from the source to the target before starting ongoing replication. This ensures both databases are synchronized before Change Data Capture (CDC) begins. GoldenGate Studio offers different transfer mediums for performing the initial load. These transfer mediums are:

1. **Database Link**
   The Database Link method transfers data directly from the source database to the target database using an established database link, without any intermediate storage. This method works best when both databases are in the same network or connected through a stable, high speed link. Since it uses SQL INSERT statements to move data, the process is straightforward but can be slower for large datasets due to SQL execution overhead. Database Link is less suited for large datasets or cross-region migrations, where latency could significantly slow down the process. It is best used for small to medium sized transfers where simplicity outweighs performance concerns.

2. **Object Storage**
   The Object Storage method stages initial load data in an Oracle Object Storage before importing it into the target database. GoldenGate first exports data from the source into Data Pump files, stores them in the specified bucket, and then imports them into the target. This approach eliminates the need for a direct network connection between source and target, which is suitable for cross region, cloud to cloud, or hybrid cloud migrations.

   A direct database connection may be slow or restricted, so the data is staged in Object Storage and securely fetched by the target system. This method is highly reliable for large datasets and supports resumable transfers in case of interruptions. However, it requires additional configuration for bucket creation and wallet authentication, as well as extra storage space for staging files. It is best used when the migration involves large volumes of data across different regions or environments.

3. **File Storage**
   The File Storage method facilitates initial load by exporting source data into disk files on the GoldenGate server. These files are then ingested into the target database. This method is particularly effective when both source and target databases are non Autonomous AI Database and can access a shared storage layer, such as:

   - Network File System (NFS) mounts
   - Oracle File Storage Service (FSS)
   - Docker volume mapping (for containerized database installations)

   By leveraging shared storage, whether on premises or cloud based organizations can achieve high throughput data transfer and maintain control over the staging environment.

   This method mirrors the Object Storage approach but is tailored for environments without access to native cloud object storage. By using shared storage, organizations can replicate the reliability and efficiency of object storage in on premises or hybrid setups.

   GoldenGate writes extracted data files to the shared location, and the target database reads these files for ingestion. This intermediate staging allows for data validation, auditing, and performance tuning before final load. It is particularly useful in secure environments with strict compliance requirements and high speed LAN connectivity.

   This setup ensures fast, secure data transfer, audit-friendly staging, and full control over sensitive financial data.

   To know more about intial load processing, refer to the Precise Instantiation for Oracle Using Initial Load Extract and About Data Replication Components in Oracle GoldenGate topics in *Oracle GoldenGate Microservices Documentation*.

Initial Load Support per Database

Different databases support different options for performing the initial load.

The table below outlines which initial load methods are supported for Oracle and MySQL source databases.

Initial Load Option	Oracle	MySQL
DBLink	Yes	No
File Storage	Yes	Yes
Object Storage	Yes	Yes

Support Matrix – Oracle Database Flavors and Initial Load Methods

Learn about the supported Initial Load methods for different Oracle AI Database with recommendations.

The matrix below outlines the supported Initial Load methods in GoldenGate Studio for various Oracle AI Database environments, including Oracle Autonomous AI Database and non Autonomous AI Database. These methods applies to all Recipes, including One-way, Active-active, and ZeroETL Recipes. Non Autonomous AI Database will contain both On-premise and DBaaS (Database as a service which is in Oracle Cloud Infrastructure).

Oracle Databases	Data Pump via Database Link	Data Pump using File Storage	Data Pump using Object Storage	Recommendation
Non Autonomous AI Database to non Autonomous AI Database	Yes	Yes	Yes	Choose Database Link for small datasets; Object Storage for large migrations.
Autonomous AI Database to non Autonomous AI Database	Yes	No	Yes	Use Object Storage for secure and reliable transfer.
Non Autonomous AI Database toAutonomous AI Database	Yes (Private Network)	No	Yes	Prefer Object Storage; use Database Link only if private network connectivity is available.
Autonomous AI Database to Autonomous AI Database	Yes	No	Yes	Object Storage is recommended for cross-region or large dataset transfers.

Extract Advanced Options

Explore the Extract Advanced Options for Oracle and MySQL database.

Setting	Description	Values
Source Database timezone	Specifies the time zone of the source database. This is a critical setting for ensuring accurate timestamp replication.	`EST,PST,UTC,IST`
Additional Extract parameters	This parameter instructs the Replicat process to ignore the records that encounter error and continue processing. These are custom parameters that can be used to handle specific error codes and conditions.	`REPERROR (PROCEDURE, DISCARD)`

If **Extract auto restart** option is enabled, you will see the following options:

Setting	Description	Values
Max Retries	The maximum number of times the Extract process will attempt to restart after a failure, before it gives up.	`0,1,2,3,4,5`
Retry Delay	The time interval, in seconds, between each restart attempt.	`1s,2s,3s,4s`

Restart Window	The time frame within which the `Max Retries` count is applied.	`1m`, `2m`, `3m`

> **Note**
>
> The **Restart on failure only** option is available in this release. When enabled, the process restarts only after a failure, not for normal stops.

Failures	The total number of times the Extract task has failed within the specified Restart Window.	`0,1,2,3,4,5`

> **Note**
>
> The **Disable task after retries exhausted** option is available in this release. When enabled, the task is automatically disabled if the maximum retry attempts are reached without success.

To enable auto-start option for Extract, you can use the following parameters:

Parameter	Description	Default Value
`extractAutostartEnable`	Enables auto start for extract process	`false`

Parameter	Description	Default Value
`extractAutostartDelay`	Delay before starting extract process	`5s`

Replicat Advanced Options

Explore the Replicat Advanced Options.

Setting	Description	Value
Action upon DML Error	Specifies the action to take when a Data Manipulation Language (DML) operation fails.	DISCARD, IGNORE, KILL, DEFAULT(RETRY_OPERATION), ABORT_TRANSACTION
Maximum Retry Count	The maximum number of tries can be specified by the users.	0, 1, 2, 3, 4, 5
Action upon DDL Error	Specifies the action to take when a Data Definition Language (DDL) operation fails.	DISCARD, IGNORE, KILL
Delay for aborting the transaction	The amount of time the system will wait before forcefully aborting open transactions that have not completed.	1h
Additional Replicat Parameters	This parameter instructs the Replicat process to ignore records that encounter Oracle error and continue processing. These are custom parameters that can be used to handle specific error codes and conditions.	REPERROR (26961, DISCARD)

If Replicat Auto Restart option is enabled, you will see the following options:

Setting	Description	Values
Retry Delay	The amount of time (in sec) to pause between discovering that the process has terminated abnormally and restarting the process.	0, 1, 2, 3, 4, 5
Failure	The number of times a task or process has failed in the current monitoring window.	0, 1, 2, 3, 4, 5
Max Retries	The maximum number of times the Replicat process will attempt to restart after a failure. This setting works in conjunction with RETRYDELAY.	0, 1, 2, 3, 4, 5
Restart Windows	Defines the time frame within which the Max Retries count is applied.	1m, 2m, 3m

To configure auto start for Replicat, you can use the following parameters:

Parameter	Description	Default Value
replicatAutostartEnable	Enables auto start for the Replicat process	false
replicatAutostartDelay	Delay before starting the Replicat process	5s

4

Manage

Learn to manage GoldenGate Studio deployments, connections, and pipelines.

Manage Deployments

Manage deployments registered in GoldenGate Studio.

Edit Deployments

Learn how to edit deployments in Oracle GoldenGate Studio's user interface.

1. In the navigation menu, select **Deployments**.
2. On the Deployments page, select **Edit** from the Actions menu of the deployment you want to modify.
3. From the **Edit Deployment** dialog box, you can edit the following options for a deployment:
   - **Name**: Name of the deployment.
   - **Hostname**: Hostname or IP address used to connect to the deployment.
   - **Port**: Port number of the Administration Service for the specified deployment.
   - **Username**: User name of the deployment to be connected. This user is the Oracle GoldenGate administrator user. See Authorization in Oracle GoldenGate
   - **Password**: Password of the user account associated with the deployment login credentials.
   - **Use Reverse Proxy**: Enable this option if reverse proxy configuration was implmented for the deployment in Oracle GoldenGate.
4. Click **Test Connection** to check if the deployment connects successfully. If the deployment connects successfully, then the **Next** button is activated.
5. Click **Next**. The Confirmation screend displays the deployment details. You can review the details and if you accept the deployment credentials, click **Save**.

Test a Deployment Connection

Testing the connection ensures that GoldenGate Studio and Oracle GoldenGate can communication with each other.

You can test a deployment in two ways:

- When you first register the deployment
- When you edit the deployment

After reviewing the deployment connection information, click **Test connection**.

Delete a Deployment

Delete deployments you no longer use.

Before you delete a deployment, check that the deployment is not used in any active pipelines.

To delete a deployment:

1. On the Deployments page, you can select the deployment to delete.

   You can also select **Delete** from the deployment's Actions menu (ellipsis).
2. On the Deployment's details page, click **Delete**.
3. In the Delete Selection dialog, confirm that this is the deployment you want to delete, and then click **Delete**.

Manage Connections

Manage connections from the GoldenGate Studio user interface including steps to edit, clone, and delete connections.

View Connection Details

Select a connection from the Database connections page to view its details. On the Connection details page, you can:

- View connection details such as:
  - Connection name, description, when it was created, and when it was last updated.
  - Database type, hostname, port, username, connector, and connection type.
  - Connection role (source or target).
- Edit the connection details.
- Validate the connection details.
- Clone the connection.
- Delete the connection.

Clone a Connection

Check the steps how to clone the database connection.

To clone a connection:

1. On the **Database Connections** page, select the database connection to clone.
2. On the **Connections** page, click **Clone**.
3. Click **Save changes**.

Edit a Connection

To edit a connection:

1. On the **Connections** page, select a database connection to edit. Click **Edit**

2. In the **Edit Connection** panel, review the connection settings, make your updates, and then click **Next** to progress though the **Edit** connection pages. You can edit details like **Database connection name, description, Hostname, Port, Username, Password, Pluggable Database Service Name.**
3. Click **Save changes**.

Delete a connection

Before you delete a connection, ensure that the connection is not currently in use by an active pipeline.

To delete a connection:

1. On the Database connections page, select the connection to delete. You're brought to the connection details page.
2. On the connection details page, click **Delete**.
3. In the Delete connection dialog, confirm that you want to delete the connection, and then click **Delete**.

The Database connections list is refreshed and the deleted connection removed. You cannot undo a delete operation.

Test a Connection

Learn why testing the connection is essential while creating the database connection.

Before you can design, deploy, or manage data replication solutions in Oracle GoldenGate Studio, it's essential to verify that your configured database connections are functioning correctly. Testing the database connection ensures that GoldenGate Studio can successfully communicate with your source and target databases using the specified configuration and credentials.

This step helps identify potential connectivity issues early, such as incorrect network settings, authentication problems, or configuration errors, thereby reducing troubleshooting time and ensuring a smooth development and deployment process.

Test Oracle Database and Exadata Database Connection

Follow these simple steps to test Oracle AI Databaseand Exadata Database Connection.

1. Select **Oracle AI Database** or **Exadata** as the **Technology Type.**
2. The next screen will show fields for username, password, hostname, port, and database Type. Enter the **Hostname** and **Port Number** for your Exadata database.
3. Select the **Database Type** and provide the **Pluggable database service name (PDB)** if applicable.
4. Click **Test Connection** to confirm if the connection is successfully working.

Test Oracle Autonomous AI Database Connection

Follow these simple steps to test Oracle Autonomous AI Database connection.

1. In Home page click **Create Connection**. Fill in connection name, optional description and **Technology Types**.
2. Select **Oracle Autonomous AI Database** as the **Technology Type**.

3. The next screen will prompt you for the **Wallet details**. You will need to provide a username and password. You also need to add the **Wallet File** that contains the credentials and connection information of your Autonomous Database.
4. After filling in the details, click **Test Connection** to verify the connection.

Test MySQL Database Connection

Follow these simple steps to test MySQL Database connection.

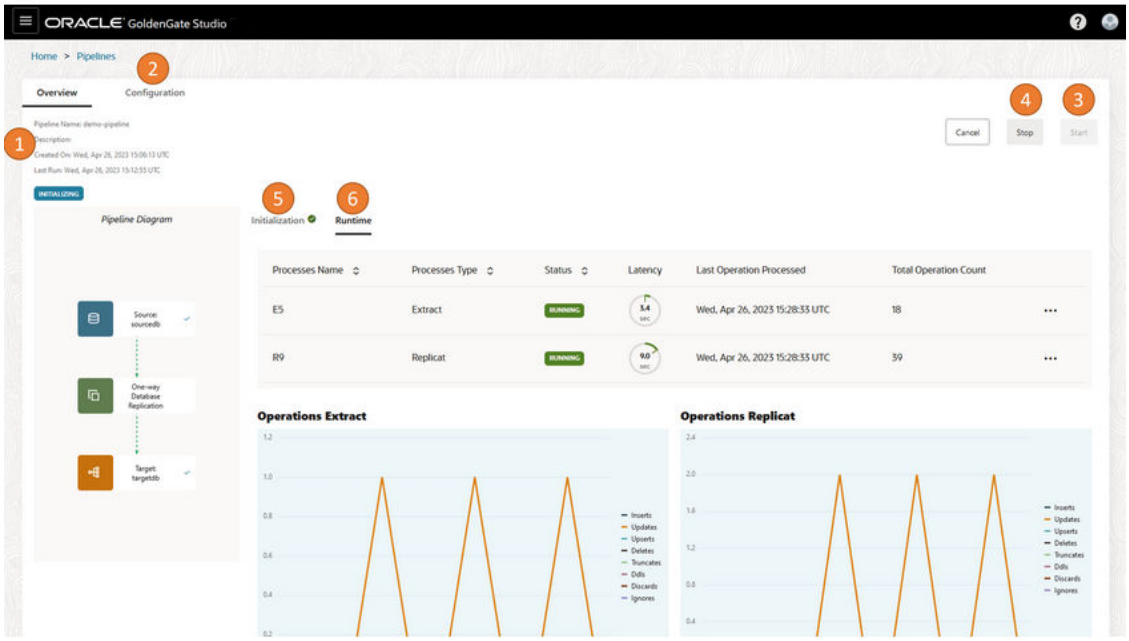
- 1. In Home page, click **Create Connection**. Fill in connection name, optional description and technology type. Select any MySQL Database, from the **Technology Type** dropdown.
  2. On the Connection Details page, complete the fields as follows::
     - Enter **Username** and **Password** for the GoldenGate admin user.
     - Enter the **Hostname**, **Port**, and **Database Name**.
     - Enter SSL Details. From the **Security Protocol** dropdown menu, select one of the following options: **plain** or **TLS**.

     then click **Next**.
  3. Click **Test Connection** to confirm if the connection is successfully working.

Manage Pipelines

Manage pipelines from the GoldenGate Studio user interface including steps to edit, start, stop, and delete pipelines.

View Pipeline Details



Follow these steps to view pipelines details:

On the **Pipelines** page, select a pipeline to view its details. On the pipeline's details page, click ellipsis (three dots) icon and then click **View.** You can:

- Review pipeline details such as the pipeline name, description, when it was created, when it was last run, and whether it ran successfully.
- Edit the pipeline's configuration, mappings, mapping rules, and advanced options.
- Start and stop the pipeline.
- View the pipeline as it runs in realtime.

> **Note**
>
> The pipeline runs continuously until you click Stop.

- View its initialization steps, the status and logs for each step, and initial load details.

> **Note**
>
> To view message logs for a step, on the **Overview** page, click the ellipsis (three dots) icon, and then select **Log Events**.

Start a Pipeline

Before you can start a pipeline, you must first Configure a Pipeline

To start a pipeline:

1. On the **Pipelines** page, select the pipeline to start.
2. On the pipeline's details page, click **Start**.

   You're brought to the pipeline's initialization page, where you can observe the initial load tasks performed on the target database, as well as the creation and start of Extract and Replicat processes.
3. After the pipeline is initialized, you're brought to the Runtime page, where you can observe data capture operations on the source database replicated to the target database.

The pipeline runs continuously until it's stopped.

Stop a Pipeline

You can only stop a pipeline that is running. Pipelines run continuously until you manually stop them. When you stop a pipeline, GoldenGate Studio stops both the Extract and Replicat processes.

To stop a pipeline:

1. On the Pipelines page, select the pipeline to stop. You're brought to the pipeline's details page.
2. On the pipeline's details page, click **Stop**.

After the pipeline stops, you can click **Start** to restart it. GoldenGate Studio resumes the Extract and Replicat processes, replicating any transactions that occurred while the pipeline was stopped.

Delete a Pipeline

To delete a pipeline:

1. On the Pipelines page, from the Action menu (ellipsis icon) of the pipeline you want to delete, select **Delete**.
2. In the Delete dialog, select whether you want to also delete the GoldenGate configuration.

> **Note**
>
> GoldenGate configuration includes the Extract and Replicat processes created when the pipeline was originally initialized.

3. Click **Delete**.

The Pipeline page refreshes, and the deleted pipeline no longer appears in the list. Once deleted, you can't undo the action.

Manage Users

GoldenGate Studio currently only supports a single user account. Learn how to change the user password.

To change the GoldenGate Studio user account password:

1. Stop the server. Use the following command to identify the server process ID:

```
netstat -tulpn | grep <server port>
```

2. Use the following command to kill the proccess:

```
kill <process-id>
```

3. Use the following command to change the user password:

```
./setPassword.sh
```

4. Enter the username, `oggadmin`, when prompted, and then press Enter.
5. Enter the new password when prompted, and then press Enter.
6. Use the following command to restart the server.

```
./run-studio.sh
```

5

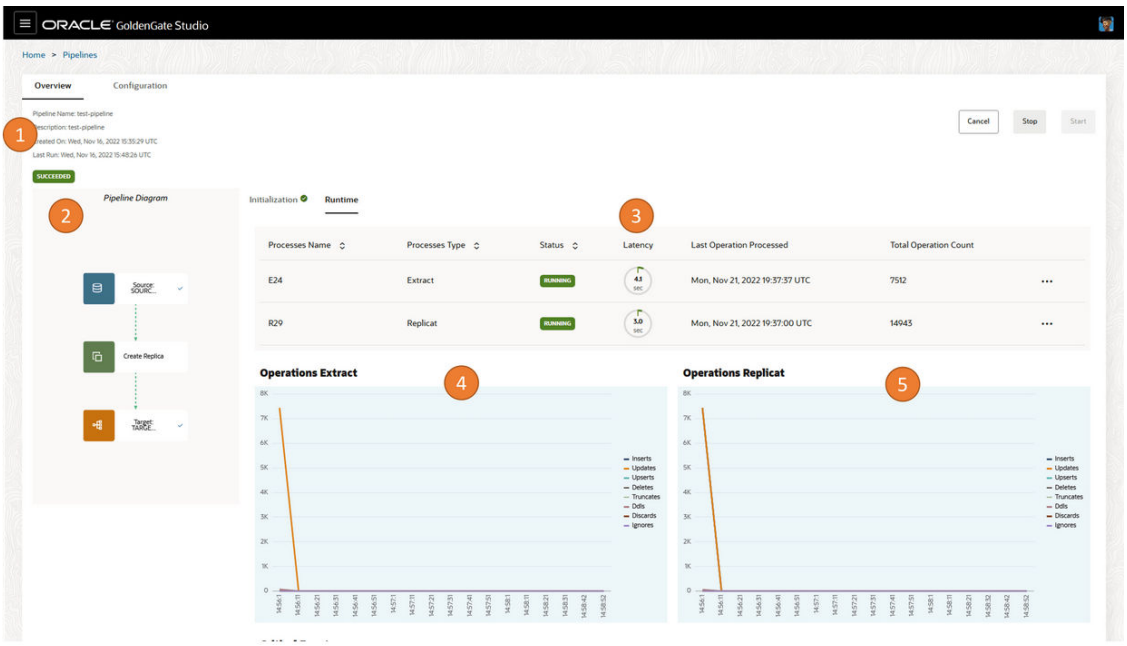
Monitor

Learn to monitor Oracle GoldenGate Studio processes and how to use the information reported.

GoldenGate Studio provides a graphical interface for designing, deploying, and managing data replication solutions. Monitoring pipelines is a critical aspect of ensuring your data integration projects run efficiently and reliably.

Monitor Pipelines

Monitor your pipelines to ensure that your data replication processes are running smoothly without lag. Use the tools available to troubleshoot or diagnose issues you may encounter.



When you select an active pipeline from the **Pipelines** page, the pipeline's **Runtime** page appears. This page displays the following details:

1. Basic pipeline information, including the pipeline's name, description, created date, and run date.
2. A real-time visual pipeline diagram, that updates as you make changes to the pipeline configuration.
3. Information about the pipeline processes, including process names, process types, their status, their latency, when their last operation was processed, and their total operation count.
4. Log events, reports, latency graphs of processes.

5. Operations' Extract graph, showing inserts, updates, upserts, deletes, truncates, DDLs, discards, and ignores over time.
6. Operations' Replicat graph, showing inserts, updates, upserts, deletes, truncates, DDLs, discards, and ignores over time.
7. (Not shown) A list of critical events, along with their codes, when they occurred, their severity, and message details.

   For each process, you can access the following when you open the ellipsis (three dots) menu:

- View log events
- Access the Oracle GoldenGate Administration Service web interface
- Download reports
- Download latency details

6
Troubleshoot

Learn about various troubleshooting tasks during advanced operations.

Configure Log Levels for Server and Performance Logs

GoldenGate Studio allows you to set up different log levels when accessing the GoldenGate Studio Server and Performance logs using logging properties.

Perform the following steps To enable logging at different levels:

1. Before setting up the log levels using the .properties files, you must set up the `LOGGER_FORMAT` environment variable. This environment variable is required for the system to determine the format in which the log needs to be generated. You can specify the value of this variable as `TEXT` or `JSON`, as needed.

   Run the following command to set the environment variable:

   ```
   export LOGGER_FORMAT=JSON/TEXT
   ```

2. Navigate to the *`studio_install_path`*`/config/logging.properties` for `TEXT` format or the *`studio_install_path`*`/config/logging-json.properties` for `JSON` format.
3. Edit the following properties in the `.properties` file:

   ```
   .level=FINE
   java.util.logging.FileHandler.level=FINE
   oracle.cloud.ggfe.log.PerformanceFileHandler.level=FINE
   oracle.cloud.ggfe.level=FINE
   oracle.cloud.ggfe.orchestrator.connectors.gg.services.level=FINE
   ```

   You can also change the log level for specific classes or packages. The following examples show the modification of log levels for the Helidon and third-party libraries:

   Example 1:

   ```
   # Quiet tracing
   io.helidon.microprofile.tracing.level=SEVERE
   io.helidon.microprofile.openapi.level=SEVERE
   io.helidon.tracing.tracerresolver.level=SEVERE
   ```

   Example 2:

   ```
   # Persistence and Jersey
   org.eclipse.persistence.level=INFO
   org.glassfish.jersey=INFO
   org.glassfish.jersey.server.level=INFO
   ```

Retrieve the Initial Load Log Files

Retrieve the initial load log files depending on the transfer medium and type of database.

The following table describes how to determine the location of the initial load log files and access them, depending on the type of database and the transfer medium:

Transfer Medium	Database	Action
Database Link	Target database, non-Oracle Autonomous Database	1. Use the following query to locate the `IMPDAT*.log` file from the `ORACLE_BASE` directory of target database. `SELECT directory_path FROM dba_directories WHERE directory_name='ORACLE_BASE'` 2. View or Copy `IMPDAT*.log` files from the `ORACLE_BASE` directory of target database.
Local Storage	Source database	1. Use the following query to locate the `EXPDAT*.log` file from the `ORACLE_BASE` directory of the source database. `SELECT directory_path FROM dba_directories WHERE directory_name='ORACLE_BASE'` 2. View or Copy `EXPDAT*.log` files from the `ORACLE_BASE` directory.

Transfer Medium	Database	Action
Local Storage	Target database	1. Use the following query to locate the `IMPDAT*.log` file from the `ORACLE_BASE` directory of target database. `SELECT directory_path FROM dba_directories WHERE directory_name='ORACLE_BASE'` 2. Copy `IMPDAT*.log` files from the `ORACLE_BASE` directory of target database.
Object Storage	Source database, Non-Oracle Autonomous Database	1. Use the following query to locate the `EXPDAT*.log` file from the `ORACLE_BASE` directory of the source database. `SELECT directory_path FROM dba_directories WHERE directory_name='ORACLE_BASE'` 2. Download `EXPDAT*.log` from `ORACLE_BASE` directory.
Object Storage	Source Database, Oracle Autonomous Database	Download `EXPDAT*.log` from the Bucket URI.
Object Storage	Target databsae, non-Oracle Autonomous Database	1. Use the following query to locate the `IMPDAT*.log` file from the `ORACLE_BASE` directory of target database. `SELECT directory_path FROM dba_directories WHERE directory_name='ORACLE_BASE'` 2. Download `IMPDAT*.log` from the `ORACLE_BASE` directory.