

Oracle® Database

Upgrading Oracle GoldenGate



21c (21.3.0)

F41985-02

January 2023

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

Oracle Database Upgrading Oracle GoldenGate, 21c (21.3.0)

F41985-02

Copyright © 2013, 2023, Oracle and/or its affiliates.

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

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

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

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

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

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

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

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

Contents

Audience

Audience	iv
Documentation Accessibility	iv
Related Information	iv
Conventions	v

Part I Upgrading Oracle GoldenGate Classic

A Overview of the convprm Tool

A.1 Running convprm	A-1
---------------------	-----

B Upgrading from Oracle GoldenGate 11.2.0.1

C Oracle Database Net Service

Audience

This guide is intended for users responsible for installing and managing Oracle GoldenGate Classic Architecture for Oracle and various non-Oracle databases.

- [Audience](#)
- [Documentation Accessibility](#)
- [Related Information](#)
- [Conventions](#)

Audience

This guide is intended for system and database administrators who are responsible for operating Oracle GoldenGate and maintaining its performance.

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>.

Accessible Access to Oracle Support

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

Related Information

The Oracle GoldenGate Product Documentation Libraries are found at:

[Oracle GoldenGate Documentation](#)

Oracle GoldenGate for Big Data Documentation:

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

For additional information on Oracle GoldenGate, refer to:

<https://www.oracle.com/middleware/technologies/goldengate.html>

[Oracle Database High Availability](#)

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, such as "From the File menu, select Save ." Boldface also is used for terms defined in text or in the glossary.
<i>italic</i> <i>italic</i>	Italic type indicates placeholder variables for which you supply particular values, such as in the parameter statement: <code>TABLE <i>table_name</i></code> . Italic type also is used for book titles and emphasis.
monospace MONOSPACE	Monospace type indicates code components such as user exits and scripts; the names of files and database objects; URL paths; and input and output text that appears on the screen. Uppercase monospace type is generally used to represent the names of Oracle GoldenGate parameters, commands, and user-configurable functions, as well as SQL commands and keywords.
UPPERCASE	Uppercase in the regular text font indicates the name of a utility unless the name is intended to be a specific case.
{ }	Braces within syntax enclose a set of options that are separated by pipe symbols, one of which must be selected, for example: <code>{<i>option1</i> <i>option2</i> <i>option3</i>}</code> .
[]	Brackets within syntax indicate an optional element. For example in this syntax, the <code>SAVE</code> clause is optional: <code>CLEANUP REPLICAT <i>group_name</i> [, SAVE <i>count</i>]</code> . Multiple options within an optional element are separated by a pipe symbol, for example: <code>[<i>option1</i> <i>option2</i>]</code> .

Part I

Upgrading Oracle GoldenGate Classic

Learn how to upgrade Oracle GoldenGate Classic from previous releases of Oracle GoldenGate.

Topics:

- [Prerequisites](#)
- [Oracle GoldenGate Upgrade Considerations](#)
- [Upgrading Oracle GoldenGate Classic Architecture](#)
- [Upgrading Oracle GoldenGate for Heterogeneous Databases](#)

Prerequisites

Before performing the upgrade procedure, read the *Release Notes for Oracle GoldenGate* to determine whether the new release affects the following in your configuration:

- New default process behavior.
- Parameters that changed or were deprecated.
- Parameters that were added to support a desired new feature or database type.
- Parameter default values that have changed.
- New data type support that might require changes to `TABLE` or `MAP` statements.
- Interaction with native database components that might require database change.

As a best practice, perform a minimal upgrade first, so that you can troubleshoot more easily in the event that any problems arise. When you verify that your environment is upgraded successfully, you can implement the new functionality.

When upgrading your database and Oracle GoldenGate simultaneously, you must upgrade the database first. However, ensure that the Oracle GoldenGate version is equal to or higher than the database version.

For Oracle Database, if you are using symbolic links that point to the Oracle GoldenGate directories, such as `dirprm` and `dirdat`, you need to use the parameter `ALLOWOUTPUTDIR` within `GLOBALS`.

You can prevent startup delays that can cause lag by having any required parameter changes made ahead of time, so that they are ready when you restart the processes. You should not make parameter changes while a process is running, but you can:

1. Make a copy of the parameter file.
2. Make edits to the copy.
3. After you shut down the processes during the upgrade procedure, copy the old parameter file to a new name (to save it as backup).
4. Copy the new parameter file to the old parameter file's name.

Take into account the following pre-upgrade requirements:

- Only if the trail file format is being changed, allow the Oracle GoldenGate processes to finish processing all current DML and DDL data in the Oracle GoldenGate trails.
- When upgrading your database and Oracle GoldenGate simultaneously, you must upgrade the database first.

Oracle GoldenGate Upgrade Considerations

Before you start the upgrade, review the information about upgrading Oracle GoldenGate Extract and Replicat for Classic Architecture.

Even though you may only be upgrading the source or target, rather than both, all processes are involved in the upgrade. All processes must be stopped in the correct order for the upgrade, regardless of which component you upgrade, and the trails must be processed until empty.

In a unidirectional environment, it is a best practice to upgrade the target before upgrading the source.

- [Extract Upgrade Considerations](#)
- [Replicat Upgrade Considerations](#)
- [Upgrade Considerations if Using Character-Set Conversion](#)
- [Upgrade Considerations if Using Quoted Object Names](#)

Extract Upgrade Considerations

Running Extract in classic mode (non-integrated Extract) with the Oracle database has been desupported. Before upgrading Extracts running in classic mode (non-integrated Extract) with the Oracle database, you need to upgrade the Extracts to run in integrated mode.

Upgrading Classic Extract to Integrated Extract for Oracle

As classic Extract is desupported, you must upgrade the classic Extract to an integrated Extract. Stop the classic Extract, which were configured with DDL capture, and remove the DDL triggers prior to upgrading to an integrated Extract.

Remove the DDL triggers:

- `ddl_disable`
- `ddl_remove`
- `marker_remove`

To upgrade to Extract, use the following command:

```
ALTER EXTRACT group_name to UPGRADE INTEGRATED TRANLOG
```

Also, see Switching Extract from Classic Mode to Integrated Mode in *Administering Oracle GoldenGate*.

Replicat Upgrade Considerations

All Replicat installations should be upgraded at the same time. It is critical to ensure that all trails leading to all Replicat groups on all target systems are processed until empty, according to the upgrade instructions.

When upgrading from releases prior to 19c release of Oracle GoldenGate, ensure that you do not use the `SOURCEDEF` parameter in Replicat, otherwise the Replicat will abend. However, if the trail file format is pre-12.2, then `SOURCEDEF` is still required because no metadata exists in the trail file.

Because the `TIMEZONE` datatype is managed differently with Oracle GoldenGate 21c, you may need to run the `ALTER REPLICAT extseqno` command to synchronize with newer trail files after consuming the old trail file written by the Extract.

Upgrade Considerations if Using Character-Set Conversion

Both the `TRAILCHARSET` and `SOURCEDEF` parameters are deprecated because Extract writes the source database character set with the column level. By default, these parameters are ignored but use the database character set and column character set from the table metadata. `SOURCECHARSET` parameter is only required when you need to override the source database character set and must use it with the `OVERRIDE` option.

Upgrade Considerations if Using Quoted Object Names

Oracle GoldenGate treats strings that are within single quotes as literals. Oracle GoldenGate has supported double-quoted object names since release 11.2 but did not fully implement the rule of single quotes for literals until release 12.1. Supporting double quotes for object names and single quotes for literals brings Oracle GoldenGate into compliance with SQL-92 rules and is now enabled by default. The `USEANSISQLQUOTES` parameter, which forced the SQL-92 standard in previous releases, is now deprecated.

The change to default SQL-92 rules affects object names in parameter files, input to `SQLEXEC` clauses, `OBEY` files, conversion functions, user exits, and commands. You have the following options as a result of this change:

- **Retain non-SQL-92 quote rules:** Oracle GoldenGate retains backward compatibility to enable the retention of current parameter files that do not conform to SQL-92 rules. To retain non-SQL-92 rules, add the `NOUSEANSISQLQUOTES` parameter to the `GLOBALS` file before you perform the upgrade and retain that parameter going forward. `NOUSEANSISQLQUOTES` affects Extract, Replicat, DEFGEN, and GGSCI.
- **Upgrade your parameter files to use SQL-92 rules:** Oracle GoldenGate provides the `convprm` conversion tool which you can run to convert your parameter files to be in conformance with SQL-92 rules. Run the `convprm` tool before you start the upgrade process. See [Overview of the convprm Tool](#) to know more.

Upgrading Oracle GoldenGate Classic Architecture

These instructions are for upgrading to Oracle GoldenGate Classic Architecture for Oracle databases.

Topics:

- [Obtaining the Oracle GoldenGate Distribution](#)
- [Upgrading Oracle GoldenGate Classic Architecture for Oracle Database](#)
- [Upgrading Oracle GoldenGate from OUI](#)
- [Upgrading Oracle GoldenGate using OUI – Silent](#)
- [Upgrading Configuration that includes Berkeley Database - Oracle GoldenGate 12.2 or later](#)

Obtaining the Oracle GoldenGate Distribution

To obtain Oracle GoldenGate, follow these steps:

1. Go to edelivery: edelivery.oracle.com

Also see MOS note 1645495.1 and 2193391.1 for more information.

To access Oracle Technology Network, go to <https://www.oracle.com/middleware/technologies/goldengate.html>

2. Find the Oracle GoldenGate 21c release and download the ZIP file onto your system.

For more information about locating and downloading Oracle Fusion Middleware products, see the [Oracle Fusion Middleware Download, Installation, and Configuration Readme Files](#) on Oracle Technology Network.

Upgrading Oracle GoldenGate Classic Architecture for Oracle Database

These instructions contain the procedure for performing the minimal upgrade.

Before proceeding, make certain that you have reviewed the following preparation information:

- [Prerequisites](#)
- [Oracle GoldenGate Upgrade Considerations](#)



Note:

Trigger-based DDL capture has been desupported from 21c onward, so you need to upgrade to native DDL capture.

 **Note:**

If you are using integrated capture and plan to upgrade from trigger-based DDL capture to new native DDL capture, **do not** remove the DDL trigger until prompted. Extract needs to mine DDL to the point where the redo `COMPATIBLE` level. For example, if Extract is behind by a week when the database is upgraded, Extract does not immediately switch to native DDL capture. It must be allowed to process the previous redo first, then Extract upgrades to native DDL capture automatically.

1. Use the following command in GGSCI to determine the oldest archive log that you might need to restore when Extract starts. The `Recovery Checkpoint` field shows the oldest log needed for recovery.

```
GGSCI> INFO EXTRACT group, SHOWCH
```

It's best to perform upgrade activities outside of the peak hours. If there are large and long running transactions, you may consider that on the source system, the new Extract might need to start processing from the normal recovery checkpoint, rather than the bounded recovery checkpoint, if the first record of the oldest open transaction at the time that you stop Extract is in a log that is not on the system.

You have two options:

- You can restore the archives back to, and including, the one shown in the recovery checkpoint shown with `INFO EXTRACT`.
- You can clear out the long-running transactions that apply to the Extract that you are upgrading. This can be done by skipping the transactions or by forcing them to the trail as committed transactions. Skipping a transaction may cause data loss, and forcing a transaction to the trail may add unwanted data to the trail if the transaction is rolled back. To skip or force a transaction, follow these steps:
 - a. View open transactions with the following command in GGSCI. Record the transaction ID of any transaction that you want to clean up.

```
GGSCI> SEND EXTRACT group, SHOWTRANS
```

- b. Clean up old transactions with the `SEND EXTRACT` command, using either the `SKIPTRANS` option to skip a transaction or the `FORCETRANS` option to force a transaction in its current state to the trail as a committed transaction.

```
GGSCI> SEND EXTRACT group, {SKIPTRANS | FORCETRANS} transaction_ID  
[THREAD n] [FORCE]
```

- c. After you are finished cleaning up the long-running transactions, force a Bounded Recovery checkpoint.

```
GGSCI> SEND EXTRACT group, BR BRCHECKPOINT IMMEDIATE
```

 **Note:**

A forced checkpoint is necessary because the skipped or forced transaction is not cleaned up from the Bounded Recovery checkpoint and will be shown if `SHOWTRANS` is issued again. This is a known issue. For more information about `SEND EXTRACT`, see *Reference for Oracle GoldenGate*.

2. (Target systems) In GGSCI, stop all Replicat processes.

```
GGSCI> STOP REPLICAT group
```

3. (Source and target systems) In GGSCI, stop Manager on the source and target systems.

```
GGSCI> STOP MANAGER
```

4. When updating target systems only, or if updating the target side before the source side, you *must* use `STOP` to stop all data pumps and any primary Extracts that write directly to those targets on any source running on this target. Any static collectors that may have been started that must be stopped, as well. To verify that there are no `server` processes running, use process checking shell commands, such as `ps` and `grep`.
5. You need to use an out-of-place upgrade, which implies that you retain the existing installation in parallel while you run the upgrade. See *Installing the Classic Architecture*, *Installing Patches for Oracle Platforms*, and *Patching Oracle GoldenGate Classic for Heterogeneous Databases in Installing Oracle GoldenGate* for details.
6. In GGSCI, start the Oracle GoldenGate processes on the source and target systems in the following order.

```
GGSCI> START MANAGER
GGSCI> START EXTRACT group
GGSCI> START EXTRACT pump
GGSCI> START REPLICAT group
```

If you need to restore any log files, Extract abends with an error that indicates the log to restore. Restore the logs back to, and including that log, and then restart Extract.

If you made copies of the parameter files to make parameter changes, move the new parameter files into the Oracle GoldenGate directory where the old parameter files were stored then rename them to the same names as the old parameter files. If you are using case-sensitivity support, ensure that you either add `NOUSEANSISQLQUOTES` to your parameter files, or that you ran the `convprm` utility to convert the quotes as required. See "[Upgrade Considerations if Using Character-Set Conversion](#)" for more information.

Upgrade Considerations for Older Oracle GoldenGate Releases

- To accommodate the changes in the checkpoint table and heartbeatable, it is recommended that you upgrade the heartbeatable (source and target) and the

checkpointtable (target only). If there is no change related to the objects, then the command returns with an informational message only. This step updates the table definition to add columns that support the Oracle GoldenGate 18c (18.1.0) release.

```
GGSCI> DBLOGIN USERIDALIAS [alias] |
GGSCI> UPGRADE CHECKPOINTTABLE [owner.table]
```

In case of SQL Server and MySQL, you need to specify the `SOURCEDB` data source with `DBLOGIN`. See `DBLOGIN` command for details.

`owner.table` can be omitted if the checkpoint table was created with the name listed with `CHECKPOINTTABLE` in the `GLOBALS` file. If the checkpoint table is already upgraded, then this command doesn't perform any further updates.

Upgrading Oracle GoldenGate from OUI

You can use Oracle Universal Installer (OUI) on any of the Linux, UNIX, and Windows platforms that OUI supports and which Oracle GoldenGate supports. OUI is supported for Oracle versions 11g and later. An instance of Oracle GoldenGate can be installed for only one Oracle version in any given Oracle home. You can install multiple instances of Oracle GoldenGate for the same or different database versions on the same host.

The installer registers the Oracle GoldenGate home directory with the central inventory that is associated with the selected database. The inventory stores information about all Oracle software products installed on a host, provided the product was installed using OUI.

To perform the upgrade using OUI, perform the following steps:

1. Unzip and untar the installation file.
2. From the unzipped directory, run the **runInstaller** program on UNIX or Linux, or `run setup.exe` on Windows.
3. On the **Select Installation Option** page, select the Oracle GoldenGate build to install, and then click **Next** to continue.
4. On the **Specify Installation Details** page, specify the following:
 - For **Software Location**, specify the Oracle GoldenGate installation directory. It can be a new or existing directory. The default location is under installing user's home directory, but Oracle recommends changing it to a local directory that is not mounted and has no quotas. The specified directory cannot be a registered home in the Oracle central inventory. If installing in a cluster, install Oracle GoldenGate on shared storage that is accessible by all of the cluster nodes.
 - (Optional) Select **Start Manager** to perform configuration functions, such as creating the Oracle GoldenGate sub-directories in the installation folder, setting library paths, and starting Manager on the specified port number. To proceed, a database must exist on the system. When Start Manager is selected, the **Database Location and Manager Port** fields are displayed.
 - The database must have a registered home in the Oracle central inventory. The installer registers the Oracle GoldenGate home directory with the central inventory.
 - For Manager Port, accept the default port number or enter a different unreserved, unrestricted port number for the Manager process to use for inter-process communication. The default port is the first available one starting with 7809. If

you are installing multiple instances of Oracle GoldenGate on the same system, each must use a different port number.

5. Click **Next** to continue. In case of upgrading existing Oracle GoldenGate Installation, OUI prompts that the selected Software location has files or directories. Click on **Yes**.
6. The **Create Inventory** page is displayed if this is the first Oracle product to be installed from OUI on a host that does not have a central inventory.
 - For **Inventory Directory**, specify a directory for the central inventory. It can be a new directory or an existing directory that is empty and has the amount of disk space shown on the screen. The directory cannot be on a shared drive.
 - Select an operating system group in which the members have write permission to the inventory directory. This group is used to add inventory information to the Oracle GoldenGate sub-folder.
7. On the **Summary** page, confirm that there is enough space for the installation and that the installation selections are correct. Optionally, click **Save Response File** to save the installation information to a response file. You can run the installer from the command line with this file as input to duplicate the results of a successful installation on other systems. You can edit this file or create a new one from a template.
8. Click **Install** to begin the installation or **Back** to go back and change any input specifications. When Upgrading existing Oracle GoldenGate Installation, OUI will notify that the software location has files or directories. Click **Yes** to continue. You are notified when the installation is finished.
9. If you created a central inventory directory, you are prompted to run the `INVENTORY_LOCATION/orainstRoot.sh` script. This script must be executed as the root operating system user. This script establishes the inventory data and creates sub-directories for each installed Oracle product (in this case, Oracle GoldenGate).

Upgrading Oracle GoldenGate using OUI – Silent

These instructions apply to new installations, as well as upgrades.

You can perform a silent installation from the command console if the system has no X-Windows interface or to perform an automated installation. Silent installations can ensure that multiple users in your organization use the same installation options when they install your Oracle products.

You perform a silent installation by running a response file. You can create a response file by selecting the **Save Response File** option during an interactive OUI session or by editing a template.

1. To run a response file, use the following command:

```
runInstaller -silent -nowait -responseFile  
absolute_path_to_response_file
```

The response files and the template are stored in the response subdirectory of the Oracle GoldenGate installation directory. The Oracle GoldenGate response file contains a standard set of Oracle configuration parameters in addition to

parameters that are specific to Oracle GoldenGate. These parameters correspond to the fields in the interactive session.

 **Note:**

If you are upgrading an existing Oracle GoldenGate installation with the silent option, then you might get the following warning:

```
WARNING:OUI-10030:You have specified a non-empty directory to install
this product. It is recommended to specify either an empty or a non-
existent directory.
```

```
You may, however, choose to ignore this message if the directory contains
Operating System generated files or subdirectories like lost+found. Do
you want to proceed with installation in this Oracle Home?
```

2. Press **ENTER** to continue.

Upgrading Configuration that includes Berkeley Database - Oracle GoldenGate 12.2 or later

When you are upgrading Oracle GoldenGate from release 12.1.2.1 to 12.3.0.1 and have enabled monitoring and the datastore is created by the Performance Metrics server, the best practice is to purge the data store before performing the upgrade. After the upgrade, the datastore is recreated. For more information about purging a datastore, see *How to Purge the Datastore*, in the *Step by Step Data Replication Using Oracle GoldenGate Microservices Architecture*.

From Oracle GoldenGate 12.3.0.1 onward, all operations related to the datastore have been removed and are taken care of by the Performance Metrics server. To know more, see *Monitoring Performance*.

Upgrading Oracle GoldenGate for Heterogeneous Databases

These instructions are for upgrading Oracle GoldenGate Classic Architecture in the supported heterogeneous database environments.

Topics:

- [Obtaining the Oracle GoldenGate Distribution](#)
- [Overview of the Upgrade Procedure for Heterogeneous Databases](#)
- [Upgrading Oracle GoldenGate Classic Architecture for Heterogeneous Databases](#)

Obtaining the Oracle GoldenGate Distribution

To obtain Oracle GoldenGate, follow these steps:

1. Go to edelivery: edelivery.oracle.com

Also see MOS note 1645495.1 and 2193391.1 for more information.

To access Oracle Technology Network, go to <https://www.oracle.com/middleware/technologies/goldengate.html>

2. Find the Oracle GoldenGate 21c release and download the ZIP file onto your system.

For more information about locating and downloading Oracle Fusion Middleware products, see the [Oracle Fusion Middleware Download, Installation, and Configuration Readme Files](#) on Oracle Technology Network.

Overview of the Upgrade Procedure for Heterogeneous Databases

The upgrade performs a minimal feature upgrade to deploy only the core Oracle GoldenGate functionality, without implementing any of the major new features. It ensures easy troubleshooting of any upgrade related issues that may occur after the upgrade. After upgrading the Oracle GoldenGate environment successfully, you can implement the new functionality.

If you are upgrading multiple Extract processes that operate in a consolidated configuration (many sources to one target), you must upgrade one Extract at a time. All Replicat installations must be upgraded at the same time. It is critical to ensure that all trails leading to all Replicat groups on all target systems are processed until empty.

Note:

The hash calculation used by the @RANGE function to partition data among Replicat processes has been changed. This change is transparent, and no re-partitioning of rows in the parameter files is required. To ensure data continuity, ensure that you allow all Replicat processes on all systems to finish processing all the data in their trails before stopping those processes. If the Replicat processes are not upgraded all at the same time, or the trails are not cleaned out prior to the upgrade, rows may shift partitions as a result of the new hash method, which may result in collision errors.

Upgrading Oracle GoldenGate Classic Architecture for Heterogeneous Databases

Even though you may only be upgrading Extract or Replicat, rather than both, all processes are involved in the upgrade. All processes must be stopped in the correct order for the upgrade, regardless of which component you upgrade, and the trails must be processed until empty.

1. (Source and target systems) Back up the current Oracle GoldenGate installation directory on the source and target systems, and any working directories that you have installed on a shared drive in a cluster (if applicable).
2. (Source and target systems, as applicable) Expand the Oracle GoldenGate upgrade build into a new directory on each system (not the current Oracle GoldenGate directory). Do not create the sub-directories; just complete the steps to the point where the installation files are expanded.

However, this step doesn't apply to PostgreSQL.

Oracle GoldenGate for PostgreSQL upgrade only works if you install the latest version (version to be upgraded) in the same `$OGG_HOME` directory as the current Oracle GoldenGate version.

For PostgreSQL, Oracle GoldenGate upgrade doesn't work if you install the latest version (version to be upgraded) in a different `$OGG_HOME` directory (21.3) and repoint the new `$OGG_HOME` to the latest version.

For PostgreSQL, use the same `$OGG_HOME` as current version directory for the latest Oracle GoldenGate binary. Make sure to take backup of existing `$OGG_HOME` before beginning the upgrade.

3. Stop all user activity that generates DML and DDL on objects in your Oracle GoldenGate configuration and ensure that there are no outstanding open transactions against the database.

For SQL Server CDC Extract on a Source system, manually stop the CDC Capture job for the database.

4. (Source system) In GGSCI on the source system, issue the `SEND EXTRACT` command with the `LOGEND` option until it shows `YES`, indicating that there is no more data in the transaction log to process.

For SQL Server CDC Extract on Source system, monitor that the current read position of the Extract is no longer updating, by repeatedly running `SEND EXTRACT group STATUS` for a few seconds, and observe that the LSN value for the current read position is no longer updating.

```
GGSCI> SEND EXTRACT group LOGEND
```

5. (Source system) In GGSCI, stop Extract and data pumps.

```
GGSCI> STOP EXTRACT group
```

6. (Target systems) In GGSCI on each target system, issue the `SEND REPLICAT` command with the `STATUS` option until it shows a status of "At EOF" to indicate that it finished processing all of the data in the trail. This must be done on all target systems until all Replicat processes return "At EOF."

```
GGSCI> SEND REPLICAT group STATUS
```

7. (Target systems) In GGSCI, stop all Replicat processes.

```
GGSCI> STOP REPLICAT group
```

8. (Source and target systems) In GGSCI, stop Manager on the source and target systems and close GGSCI.

```
GGSCI> STOP MANAGER
```

9. If you want to upgrade the source or target database, or both, do so at this time according to the upgrade instructions provided for that database. Ensure that you start the databases after the upgrade, but do not permit transactions on the objects in the Oracle GoldenGate configuration.

For MySQL, if you upgrade from Oracle GoldenGate 19c (19.1.0) and the database is MySQL 5.7, then no change is required. However, if you upgrade from Oracle GoldenGate 19c (19.1.0) and the database is MySQL 8.0, then you need to perform the following steps:

- a. Enable logging of full metadata because it's mandatory for MySQL 8.0 and higher, regardless of DDL or DML replication. Logging of full metadata can be enabled by

setting the value of MySQL server variable `binlog_row_metadata` to `FULL` inside the MySQL configuration file (`my.cnf` for Linux and `my.ini` for Windows). You need to restart the database service after changing the configuration file for the settings to take effect.

- b. Run the DDL uninstall scripts to disable old DDL solutions if DDL replication was enabled previously.

The script name:

For Windows: `ddl_install.bat`

For Linux: `ddl_install.sh`

- c. To uninstall, run the following script:

```
bash$ ./ddl_install.sh uninstall mysql userid password port
```

10. (Source and target systems) Move the expanded Oracle GoldenGate files from the new directory to your existing Oracle GoldenGate directory on the source and target systems overwriting the existing files.
11. (DB2 for i) Run `ggos400install` without arguments. For an upgrade, no arguments are necessary. However, if you change the library, the old library is left on the system until you remove it.
12. (Source and target systems) Start GGSCI.
13. (Target systems, if upgrading Replicat from version 11.2.1.0.0 or earlier only) In GGSCI on each target system, issue the following commands to upgrade the Replicat checkpoint tables on those systems. This step updates the table definition.

```
GGSCI> DBLOGIN {
    [SOURCEDB data_source] |
    [, database@host:port] |USERID {/ | userid}
    [, PASSWORD password]
    [algorithm ENCRYPTKEY {keyname | DEFAULT}] |USERIDALIAS
alias [DOMAIN domain] |
    [SYSDBA | SQLID sqlid]
    [SESSIONCHARSET character_set]}
```

```
GGSCI> UPGRADE CHECKPOINTTABLE [owner.table]
```

 **Note:**

`owner.table` can be omitted if the checkpoint table was created with the name listed with `CHECKPOINTTABLE` in the `GLOBALS` file.

14. (SQL Server Oracle GoldenGate classic Extract 12c (12.3.0.1) or prior, on Source system) Run the `DELETE TRANDATA` command against any tables enabled with it and delete the heartbeat tables if they exist (`DELETE HEARTBEATTABLE`). Then run `ADD TRANDATA` again for the tables and `ADD HEARTBEATTABLE`, if previously used.

```
GGSCI> DBLOGIN {[SOURCEDB data_source] | |USERID {/ | userid}[,
PASSWORD password] |USERIDALIAS alias [DOMAIN domain]
GGSCI> DELETE TRANDATA schema.tablename
```

```
GGSCI> DELETE HEARTBEATTABLE
GGSCI> ADD HEARTBEATTABLE
GGSCI> ADD TRANDATA schema.tablename
```

15. (SQL Server CDC Extract on Source system) Run `ADD TRANDATA` again on any tables previously enabled with it, including any filter table and the `gg_heartbeat` and `gg_heartbeat_seed` tables if using the Oracle GoldenGate heartbeat implementation.

```
GGSCI> DBLOGIN {[SOURCEDB data_source] | |USERID {/ | userid}[, PASSWORD
password] |USERIDALIAS alias [DOMAIN domain]
GGSCI> ADD TRANDATA schema.tablename
```

16. (Target system) If upgrading the target Oracle GoldenGate installation that is the recipient of trails from a source system running Oracle GoldenGate prior to version 11.2.1, then add the `SOURCECHARSET` parameter to the Replicat and specify the character set of the source database.
17. (Source system) By default, after upgrading, the Extract will continue to write trail files in the version of Oracle GoldenGate prior to the upgrade. To force the Extract to write in the upgraded trail version, use the `FORMAT RELEASE` parameter in the Extract, specifying the new version, or alternately, perform an `ETROLLOVER` of the Extract and manually reposition the downstream processes to start reading at the new trail sequence.

```
{EXTRAIL | RMTTRAIL} file_name FORMAT RELEASE major.minor
```

18. If you made copies of the parameter files to make parameter changes, move the new parameter files into the Oracle GoldenGate directory where the old parameter files were stored, and give them the same names as the old parameter files. If using case-sensitivity support, make certain that you either added `NOUSEANSISQLQUOTES` to your parameter files, or that you ran the `convprm` utility to convert the quotes as required.

See "[Upgrade Considerations if Using Character-Set Conversion](#)" for more information.

19. Upgrade the heartbeat table configuration if it was previously implemented, before restarting all the processes.

```
GGSCI> DBLOGIN {[SOURCEDB data_source] | |USERID {/ | userid}[, PASSWORD
password]
|USERIDALIAS alias [DOMAIN domain]
GGSCI> UPGRADE HEARTBEATTABLE
```

20. You also need to modify the `BATCHSQL` parameter to double the value of `BATCHESPERQUEUE`. You must do this before starting Replicat.

 **Note:**

If you are upgrading from Oracle GoldenGate version 12.1 to any later version and using the `INSERTALLRECORDS` parameter, it is recommended that you should double the value of `BYTESPEERQUEUE` option of the `BATCHSQL` parameter. For example, if you are using the `BYTESPEERQUEUE` option with its default value, which is 20 MB, then increase the value to 40 MB. However, if you are not using the default value for the `BYTESPEERQUEUE` option, then double the value specified during the Oracle GoldenGate version 12.1 installation.

For example:

```
BATCHSQL BATCHESPERQUEUE 40000000
```

21. For SQL Server CDC Extract on a Source system, manually restart the CDC Capture job for the database.
22. In GGSCI, start the Oracle GoldenGate processes on the source and target systems in the following order.

```
GGSCI> START MANAGER  
GGSCI> START EXTRACT group  
GGSCI> START EXTRACT pump  
GGSCI> START REPLICAT group
```

A

Overview of the `convprm` Tool

The following describes the `convprm` tool:

- It is a command line program which can be run either manually or scripted.
- It converts string literals from double-quoted character strings to single-quoted character strings, but leaves double-quoted object names intact. It can distinguish between an object name and a string literal even when both are represented as a sequence of characters delimited with double quotes.
- It escapes quotation marks. Quotation marks must be escaped when the character that is used to delimit the string appears in the literal string itself. For example, the sentence "This character "" is a double quote" contains an escaped quote mark. The sentence 'This character " is a single quote' contains an escaped single quote mark. When converting from double quotes to single quotes, `convprm` removes one of the repeated double quotes from escaped double quotes and escapes the single quotes that are embedded in the character sequence.
- It issues a warning message if `NOUSEANSISQLQUOTES` is specified in the `GLOBALS` file. The message states that the converted parameter file is incompatible with `NOUSEANSISQLQUOTES`, but the parameter file was updated anyway.
- It recursively converts the files that are included through an `OBEY` or `INCLUDE` parameter.
- It creates a backup of the initial parameter file in the same directory as the original file. The backup has the name of the original file with the `.bak` suffix. The creation of a backup file can be disabled when you run the `convprm` tool.
- It converts the character set. The character set for the new parameter file is taken from the `CHARSET` parameter in the original parameter file. Absent that parameter, the character set is taken from the `CHARSET` parameter in the `GLOBALS` file. Absent a `GLOBALS` parameter, the new parameter file is written in the character set of the local operating system.

[Table A-1](#) provides examples of the conversion outcome.

Table A-1 Comparison of Input Requirements for [NO]USEANSISQLQUOTES

Input variable	String literal with old syntax	String literal with new syntax
Double quotes are escaped in the old syntax but not in the new syntax.	"abc""def"	'abc"def'
Single quotes are escaped in the new syntax but not in the old syntax.	"abc'def"	'abc"def'

- [Running `convprm`](#)

A.1 Running `convprm`

To use the `convprm` tool:

1. Run `convprm` with the following syntax:

```
convprm [options] input_files
```

where:

- {-h | --help} displays usage.
- {-v | --version} displays version information.
- {-i | --follow-include} recursively converts files included through an OBEY or INCLUDE parameter.
- {-n | --no-backup} does not create a copy of the original file.
- {-s | --silent} does not display status messages.
- {-q | --quotes} performs quote conversion. This is the default.
- {-d | --dry-run} does not change the parameter file or create a backup file. It only prints out what would happen as the result of the conversion.
- *input_files* is a list of the parameter files that are to be converted. Separate each file name with a white space, for example:

```
convprm [options] extfin extacct exthr
```

2. Examine the parameter file to make certain the conversion completed successfully. Status messages are displayed at the beginning, during, or at the end of the file conversion process.

On errors, the process abends in the same way as other Oracle GoldenGate executables. All error messages that cause the converter to fail are sent to the Oracle GoldenGate error log.

If you are currently using the `USEANSISQLQUOTES` parameter, you may remove it or leave it in the parameter files. It is now the default.

B

Upgrading from Oracle GoldenGate 11.2.0.1

Oracle GoldenGate follows the SQL-92 rules for column names and literals that are enclosed within single or double quote marks. If you are upgrading from Oracle GoldenGate 11.2, you must double check the parameter files for SQL-92 compatibility with the convprm program. If the GLOBALS file contains the parameter `NOUSEANSISQLQUOTES`, it needs to be removed.

C

Oracle Database Net Service

BEQUEATH Protocol

If the connection was done with `ORACLE_HOME` and `ORACLE_SID` in the older Oracle GoldenGate release, you have to use a TNS alias or TNS connection descriptor with the unified build. Connection using `BEQ` are available on TNS.

```
inst1_beq= (DESCRIPTION=(ADDRESS=(PROTOCOL=beq)
  (ENVS='ORACLE_SID=SID,LD_LIBRARY_PATH=ORACLE_HOME/
lib,ORACLE_HOME=ORACLE_HOME')
  (PROGRAM=ORACLE_HOME/bin/oracle)
  (ARGV0=oracleSID)
  (ARGS=' (DESCRIPTION=(LOCAL=YES) (ADDRESS=(PROTOCOL=beq)) )' )
)
(CONNECT_DATA=(SID=SID))
)
```

The following example shows the use `BEQ` protocol:

```
inst1_beq= (DESCRIPTION=(ADDRESS=(PROTOCOL=beq)
  (ENVS='ORACLE_SID=PROD42,LD_LIBRARY_PATH=/u01/app/oracle/product/21.0.0.0/
lib,ORACLE_HOME=/u01/app/oracle/product/21.0.0.0')
  (PROGRAM=/u01/app/oracle/product/21.0.0.0/bin/oracle)
  (ARGV0=oraclePROD42)
  (ARGS=' (DESCRIPTION=(LOCAL=YES) (ADDRESS=(PROTOCOL=beq)) )' )
)
(CONNECT_DATA=(SID=PROD42))
)
```

TWO_TASK Environment Variable

To define a default connect identifier, use the `TWO_TASK` environment variable on Linux and UNIX platforms or the local environment variable or registry entry on Microsoft Windows.

```
export TWO_TASK=inst1_beq
```

In this case, no TNS connection string is needed. If additional environment variables can be provided with the `oggca.sh` deployment of Oracle GoldenGate.