Oracle® Fusion Middleware

Installing and Configuring Oracle GoldenGate for MySQL

12*c* (12.2.0.1)
E67826-03
July 2018

ORACLE

Oracle Fusion Middleware Installing and Configuring Oracle GoldenGate for MySQL, 12*c* (12.2.0.1)

E67826-03

Contents

A Oracle GoldenGate Installed Components

Preface

Oracle GoldenGate for MySQL supports replication from a MySQL source database to a MySQL target database or to a supported database of another type to perform an initial load or change data replication.

This guide helps you get started with installing Oracle GoldenGate on a MySQL database system and performing initial setup. Refer to the other Oracle GoldenGate documentation listed in this Preface for additional information to configure, run, and manage your Oracle GoldenGate environment.

Audience

This guide is intended for installers, database administrators, and system administrators who are installing, configuring and running Oracle GoldenGate.

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 Documents

The Oracle GoldenGate documentation set includes the following components:

Windows, UNIX, and Linux Platforms

- *Installing and Configuring Oracle GoldenGate for DB2 for i*
- *Installing and Configuring Oracle GoldenGate for DB2 LUW*
- *Installing and Configuring Oracle GoldenGate for DB2 z/OS*
- *Installing and Configuring Oracle GoldenGate for Informix*
- *Installing and Configuring Oracle GoldenGate for MySQL*
- *Installing and Configuring Oracle GoldenGate for NonStop SQL/MX*
- *Installing and Configuring Oracle GoldenGate for SQL Server*
- *Installing and Configuring Oracle GoldenGate for Oracle TimesTen*

- *Installing and Configuring Oracle GoldenGate for Oracle Database*
- *Installing and Configuring Oracle GoldenGate for Sybase*
- *Installing and Configuring Oracle GoldenGate for Teradata*
- *Administering Oracle GoldenGate for Windows and UNIX*
- *Reference for Oracle GoldenGate for Windows and UNIX*
- *Logdump Reference for Oracle GoldenGate*
- *Upgrading Oracle GoldenGate for Windows and UNIX*
- *Error Messages Reference for Oracle GoldenGate for Windows and UNIX*

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.
italic *italic*	Italic type indicates placeholder variables for which you supply particular values, such as in the parameter statement: TABLE *table_name*. 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: {*option1* | *option2* | *option3*}.
[ ]	Brackets within syntax indicate an optional element. For example in this syntax, the SAVE clause is optional: CLEANUP REPLICAT *group_name* [, SAVE *count*]. Multiple options within an optional element are separated by a pipe symbol, for example: [*option1* | *option2*].

1

System Requirements and Preinstallation Instructions

This chapter contains the requirements for the system and database resources that support Oracle GoldenGate. It contains the following sections:

- Verifying Certification and System Requirements
- Operating System Requirements
- Database Requirements

1.1 Verifying Certification and System Requirements

Make sure that you are installing your product on a supported hardware or software configuration. For more information, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page.

Oracle has tested and verified the performance of your product on all certified systems and environments; whenever new certifications occur, they are added to the proper certification document right away. New certifications can occur at any time, and for this reason the certification documents are kept outside of the documentation libraries and are available on Oracle Technology Network.

1.2 Operating System Requirements

This section describes the MySQL operating system requirements. These requirements fall into one of the following categories:

- Memory Requirements
- Disk Requirements
- Temporary Disk Requirements
- Network
- Operating System Privileges
- Console
- Other Programs

1.2.1 Memory Requirements

The amount of memory that is required for Oracle GoldenGate depends on the amount of data being processed, the number of Oracle GoldenGate processes running, the amount of RAM available to Oracle GoldenGate, and the amount of disk space that is available to Oracle GoldenGate for storing pages of RAM temporarily on disk when the operating system needs to free up RAM (typically when a low watermark is reached). This temporary storage of RAM to disk is commonly known as *swapping* or *paging* (herein referred to as *swapping*). Depending on the platform, the term *swap space* can

be a swap partition, a swap file, a page file (Windows) or a shared memory segment (IBM i platforms).

Modern servers have sufficient RAM combined with sufficient swap space and memory management systems to run Oracle GoldenGate. However, increasing the amount of RAM available to Oracle GoldenGate may significantly improve its performance, as well as that of the system in general.

Typical Oracle GoldenGate installations provide RAM in multiples of gigabytes to prevent excessive swapping of RAM pages to disk. The more contention there is for RAM the more swap space that is used.

Excessive swapping to disk causes performance issues for the Extract process in particular, because it must store data from each open transaction until a commit record is received. If Oracle GoldenGate runs on the same system as the database, the amount of RAM that is available becomes critical to the performance of both.

RAM and swap usage are controlled by the operating system, not the Oracle GoldenGate processes. The Oracle GoldenGate cache manager takes advantage of the memory management functions of the operating system to ensure that the Oracle GoldenGate processes work in a sustained and efficient manner. In most cases, users need not change the default Oracle GoldenGate memory management configuration.

For more information about evaluating Oracle GoldenGate memory requirements, see the `CACHEMGR` parameter in *Reference for Oracle GoldenGate for Windows and UNIX*.

1.2.2 Disk Requirements

Assign free disk space according to the following instructions.

- To determine the size of the Oracle GoldenGate download file, view the Size column before downloading your selected build from Oracle Software Delivery Cloud. The value shown is the size of the files in compressed form. The size of the expanded Oracle GoldenGate installation directory will be significantly larger on disk.
- To install Oracle GoldenGate into a cluster environment, install the Oracle GoldenGate binaries and files as the Oracle user on a shared file system that is available to all cluster nodes. For more information, see Preparing to Install Oracle GoldenGate Within a Cluster.
- Allocate an additional 1 GB of disk space on any system that hosts Oracle GoldenGate trails, which are files that contain the working data. You may need more or less than this amount, because the space that is consumed by the trails depends on the volume of data that will be processed. See the guidelines for sizing trails in *Administering Oracle GoldenGate for Windows and UNIX*.

1.2.3 Temporary Disk Requirements

By default, Oracle GoldenGate maintains data that it swaps to disk in the `dirtmp` sub-directory of the Oracle GoldenGate installation directory. The cache manager assumes that all of the free space on the file system is available. This directory can fill up quickly if there is a large transaction volume with large transaction sizes. To prevent I/O contention and possible disk-related Extract failures, dedicate a disk to this directory. You can assign a name and size to this directory with the `CACHEDIRECTORY` option of the `CACHEMGR` parameter. The `CACHESIZE` option of `CACHEMGR` sets a soft limit for the amount of virtual memory (cache size) that is available for caching transaction data. See

Reference for Oracle GoldenGate for Windows and UNIX for the default values of these options and detailed explanations, in case system adjustments need to be made.

1.2.4 Network

Configure networking according to the following instructions:

- Configure the system to use TCP/IP services, including DNS. Oracle GoldenGate supports IPv4 and IPv6 and can operate in a system that supports one or both of these protocols.
- Configure the network with the host names or IP addresses of all systems that will be hosting Oracle GoldenGate processes and to which Oracle GoldenGate will be connecting. Host names are easier to use.
- Oracle GoldenGate requires some unreserved and unrestricted TCP/IP ports, the number of which depends on the number and types of processes in your configuration. See for details on how to configure the Manager process to handle the required ports.
- Keep a record of the ports that you assigned to Oracle GoldenGate. You will specify them with parameters when configuring the Manager process.
- Configure your firewalls to accept connections through the Oracle GoldenGate ports.

1.2.5 Operating System Privileges

Assign operating privileges according to the following instructions.

- To install on Windows, the person who installs Oracle GoldenGate must log in as Administrator.
- To install on UNIX, the person who installs Oracle GoldenGate must have read and write privileges on the Oracle GoldenGate installation directory.
- The Oracle GoldenGate Extract, Replicat, and Manager processes must operate as an operating system user that has privileges to read, write, and delete files and subdirectories in the Oracle GoldenGate directory. In addition, the Manager process requires privileges to control the other Oracle GoldenGate processes.
- Dedicate the Extract, Replicat, and Manager operating system users to Oracle GoldenGate. Sensitive information might be available to anyone who runs an Oracle GoldenGate process.

1.2.6 Console

The operating system and the command console must have the same character sets. Mismatches occur on Microsoft Windows systems, where the operating system is set to one character set, but the DOS command prompt uses a different, older DOS character set. Oracle GoldenGate uses the character set of the operating system to send information to GGSCI command output; therefore a non-matching console character set causes characters not to display correctly. You can set the character set of the console before opening a GGSCI session by using the following DOS command:

```
chcp OS_character_set
```

If the characters do not display correctly after setting the code page, try changing the console font to Lucida Console, which has an extended character set.

1.2.7 Other Programs

Before installing Oracle GoldenGate on a Windows system, install and configure the Microsoft Visual C ++ 2010 SP1 Redistributable Package. Make certain it is the SP1 version of this package, and make certain to get the correct bit version for your server. This package installs runtime components of Visual C++ Libraries. For more information, and to download this package, go to `http://www.microsoft.com`.

Oracle GoldenGate fully supports virtual machine environments created with any virtualization software on any platform. When installing Oracle GoldenGate into a virtual machine environment, select a build that matches the database and the operating system of the virtual machine, not the host system.

1.3 Database Requirements

This section describes the MySQL database requirements. These requirements fall into one of the following categories:

- Supported Version
- Database Character Set
- Database Storage Engine
- Database User for Oracle GoldenGate Processes
- Supported MySQL Data Types
- Non-Supported MySQL Data Types
- Supported Objects and Operations for MySQL
- Non-Supported Objects and Operations for MySQL

1.3.1 Supported Version

This release of Oracle GoldenGate for MySQL supports both the MySQL Enterprise and Community Editions.

1.3.2 Database Character Set

MySQL provides a facility that allows users to specify different character sets at different levels.

Level	Example
Database	`create database test charset utf8;`
Table	`create table test( id int, name char(100)) charset utf8;`
Column	`create table test ( id int, name1 char(100) charset gbk, name2 char(100) charset utf8));`

Although you can specify entirely different character sets at all these different levels, for data conversion from source (one character set) to target (another character set), Oracle GoldenGate currently supports only character sets at the database level.

Limitations of Support

- When you specify the character set of your database as utf8mb4/utf8, the default collation is `utf8mb4_unicode_ci`/`utf8_general_ci`. If you specify `collation_server=utf8mb4_bin`, the database interprets the data as binary. For example, specifying the `CHAR` column length as four means that the byte length returned is 16 (for utf8mb4) though when you try to insert data more than four bytes the target database warns that the data is too long. This is the limitation of database so Oracle GoldenGate does not support binary collation. To overcome this issue, specify `collation_server=utf8mb4_bin` when the character set is utf8mb4 and `collation_server=utf8_bin` for utf8.

1.3.3 Database Storage Engine

Requirements for the database storage engine are as follows:

- Oracle GoldenGate supports the InnoDB storage engine for a source MySQL database.
- On a target MySQL database, the Oracle GoldenGate Replicat process connects to the database through the MySQL native API.
- The NDB cluster engine is supported.

1.3.4 Database User for Oracle GoldenGate Processes

Requirements for the database user for Oracle GoldenGate processes are as follows:

- Create a database user that is dedicated to Oracle GoldenGate. It can be the same user for all of the Oracle GoldenGate processes that must connect to a database:
  - Extract (source database)
  - Replicat (target database)
  - `DEFGEN` (source or target database)
- To preserve the security of your data, and to monitor Oracle GoldenGate processing accurately, do not permit other users, applications, or processes to log on as, or operate as, the Oracle GoldenGate database user.
- Keep a record of the database users. They must be specified in the Oracle GoldenGate parameter files with the `USERID` parameter.
- The Oracle GoldenGate user requires read access to the `INFORMATION_SCHEMA` database.
- The Oracle GoldenGate user requires the following user privileges.

Table 1-1 Oracle GoldenGate database user privileges for MySQL

Privilege	Extract	Replicat
`INSERT`, `UPDATE`, `DELETE` on target tables		X

Table 1-1 (Cont.) Oracle GoldenGate database user privileges for MySQL

Privilege	Extract	Replicat
`CREATE TABLE`		X[1]
`EXECUTE`		X[2]

`SELECT ANY TABLE`
or
`SELECT ON database.table`
	X	X

1 If using the checkpoint table feature (recommended)
2 To execute stored procedures

- To capture binary log events, an Administrator must provide the following privileges to the Extract user:
  - Read and Execute permissions for the directory where the MySQL configuration file (`my.cnf`) is located
  - Read permission for the MySQL configuration file (`my.cnf`)
  - Read and Execute permissions for the directory where the binary logs are located
  - Read and Execute permission for the `tmp` directory

1.3.5 Supported MySQL Data Types

MySQL supports the following data types:

- `CHAR`
- `VARCHAR`
- `INT`
- `TINYINT`
- `SMALL INT`
- `MEDIUM INT`
- `BIG INT`
- `DECIMAL`
- `FLOAT`
- `DOUBLE`
- `DATE`
- `TIME`
- `YEAR`
- `DATETIME`
- `TIMESTAMP`
- `BINARY`
- `VARBINARY`
- `TEXT`

- TINYTEXT
- MEDIUMTEXT
- LONGTEXT
- BLOB
- TINYBLOB
- MEDIUMBLOB
- LONGBLOB
- ENUM
- BIT(M)

1.3.5.1 Limitations and Clarifications

When running Oracle GoldenGate for MySQL, be aware of the following:

- Oracle GoldenGate does not support BLOB or TEXT types when used as a primary key.
- Oracle GoldenGate supports UTF8 and UCS2 character sets. UTF8 data is converted to UTF16 by Oracle GoldenGate before writing it to the trail.
- UTF32 is not supported by Oracle GoldenGate.
- Oracle GoldenGate supports a TIME type range from 00:00:00 to 23:59:59.
- Oracle GoldenGate supports timestamp data from 0001/01/03:00:00:00 to 9999/12/31:23:59:59. If a timestamp is converted from GMT to local time, these limits also apply to the resulting timestamp. Depending on the time zone, conversion may add or subtract hours, which can cause the timestamp to exceed the lower or upper supported limit.
- Oracle GoldenGate does not support negative dates.
- The support of range and precision for floating-point numbers depends on the host machine. In general, the precision is accurate to 16 significant digits, but you should review the database documentation to determine the expected approximations. Oracle GoldenGate rounds or truncates values that exceed the supported precision.
- When you use ENUM type in non-strict sql_mode, the non-strict sql_mode does not prevent you from entering an invalid ENUM value and an error will be returned. To avoid this situation, do one of the following:
  - Use sql_mode as STRICT and restart Extract. This prevents users from entering invalid values for any of the data types. An IE user can only enter valid values for those data types.
  - Continue using non-strict sql_mode, but do not use ENUM data types.
  - Continue using non-strict sql_mode and use ENUM data types with valid values in the database. If you specify invalid values, the database will silently accept them and Extract will abend.
- To preserve transaction boundaries for a MySQL target, create or alter the target tables to the InnoDB transactional database engine instead of the MyISAM engine. MyISAM will cause Replicat records to be applied as they are received, which

does not guarantee transaction integrity even with auto-commit turned off. You cannot roll back a transaction with MyISAM.

- Extraction and replication from and to views is not supported.
- Capturing from MySQL replication target is supported so you can capture from a MySQL replication slave.

1.3.6 Non-Supported MySQL Data Types

MySQL does not support the following data types:

- The `XML`, `SET`, and Geometry data types and similar are not supported.
- There is limited support for the Interval data type.

1.3.7 Supported Objects and Operations for MySQL

Oracle GoldenGate for MySQL supports to following objects and operations:

- Oracle GoldenGate supports the extraction and replication of transactional tables.
- Oracle GoldenGate supports transactional tables up to the full row size and maximum number of columns that are supported by MySQL and the database storage engine that is being used. InnoDB supports up to 1000 columns.
- Oracle GoldenGate supports the `AUTO_INCREMENT` column attribute. The increment value is captured from the binary log by Extract and applied to the target table in a Replicat insert operation.
- Oracle GoldenGate supports the following DML operations on source and target database transactional tables:
  - Insert operation
  - Update operation (compressed included)
  - Delete operation (compressed included); cascade delete queries result in the deletion of the child of the parent operation
  - Truncate operation
- Oracle GoldenGate supports the following initial load methods from a source MySQL database to a MySQL or other target database:
  - Loading data with Replicat
  - Loading data with an Oracle GoldenGate direct load
- Oracle GoldenGate can operate concurrently with MySQL native replication.
- Oracle GoldenGate supports the `DYNSQL` feature for MySQL.
- Limitations on Automatic Heartbeat Table support are as follows:
  - Ensure that the database in which the heartbeat table is to be created already exists to avoid errors when adding the heartbeat table.
  - In the heartbeat history lag view, the information in fields like `heartbeat_received_ts`, `incoming_heartbeat_age`, and `outgoing_heartbeat_age` are shown with respect to the system time. You should ensure that the operating system time is setup with the correct and current time zone information.

1.3.8 Non-Supported Objects and Operations for MySQL

Oracle GoldenGate for MySQL does not support the following objects and operations:

- Extraction or replication of DDL (data definition language) operations.
- The Oracle GoldenGate `BATCHSQL` feature.
- Array fetching during initial load.
- The following character sets are not supported:

  ULIB_CS_ARMSCII8, /* American National 166-9 */
  ULIB_CS_GEOSTD8, /* Geogian Standard */
  ULIB_CS_KEYBCS2, /* Kemennicky MS-DOS

2
Installing Oracle GoldenGate

Learn about installing Oracle GoldenGate for MySQL for the first time.
These instructions are for installing Oracle GoldenGate for the first time. Additionally, they are for downloading the base release of a new version of Oracle GoldenGate.

To download and install subsequent patches to the base release, go to the Patches and Updates tab of My Oracle Support at:

http://support.oracle.com

To upgrade Oracle GoldenGate from one version to another, follow the upgrade instructions in .

Topics:

2.1 Overview

Installing Oracle GoldenGate installs all of the components that are required to run and manage the processing (excluding any components required from other vendors, such as drivers or libraries) and it installs the Oracle GoldenGate utilities.

2.2 Understanding and Obtaining the Oracle GoldenGate Distribution

For complete information about how to obtain Oracle Fusion Middleware software, see "Understanding and Obtaining Product Distributions" in *Planning an Installation of Oracle Fusion Middleware*.

To download the Oracle WebLogic Server and Coherence software for development or evaluation, see the following location on the Oracle Technology Network (OTN):

http://www.oracle.com/technetwork/middleware/fusion-middleware/downloads/index.html

For more information about locating and downloading Oracle Fusion Middleware products, see the *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files* on OTN.

ORIGINAL TEXT BELOW; keep file name???

To obtain Oracle GoldenGate follow these steps:

1. Go to Oracle Technology Network.
2. Find the Oracle GoldenGate 12*c* (12.2.0.1) release and download the ZIP file onto your system.

2.3 Setting Library Paths for Dynamic Builds on UNIX

Oracle GoldenGate uses shared libraries. When you install Oracle GoldenGate on a UNIX system, the following must be done before you run GGSCI or any other Oracle GoldenGate process.

If you will be running an Oracle GoldenGate program from outside the Oracle GoldenGate installation directory on a UNIX system:

- (Optional) Add the Oracle GoldenGate installation directory to the `PATH` environment variable.
- (Required) Add the Oracle GoldenGate installation directory to the shared-libraries environment variable.

For example, given an Oracle GoldenGate installation directory of `/users/ogg`, the second command in the following example requires these variables to be set:

Command	Requires GG libraries in environment variable?
`$ users/ogg > ./ggsci`	No
`$ users > ./ogg/ggsci`	Yes

To set the variables in Korn shell

```
PATH=installation_directory:$PATH
export PATH
shared_libraries_variable=absolute_path_of_installation_directory:$shared_librarie
s_variable
export shared_libraries_variable
```

To set the variables in Bourne shell

```
export PATH=installation_directory:$PATH
export
shared_libraries_variable=absolute_path_of_installation_directory:$shared_libraries
_variable
```

To set the variables in C shell

```
setenv PATH installation_directory:$PATH
setenv shared_libraries_variable
absolute_path_of_installation_directory:$shared_libraries_variable
```

Where *shared_libraries_variable* is one of the following:

Table 2-1 UNIX/Linux library path variables per platform

Platform[1]	Environment variable
IBM AIX IBM z/OS	`LIBPATH`
HP-UX	`SHLIB_PATH`
Sun Solaris LINUX	`LD_LIBRARY_PATH`[2]

[1] A specific platform may or may not be supported by Oracle GoldenGate for your database.

[2] In 64-bit environments with 32-bit Oracle databases, Oracle GoldenGate requires the `LD_LIBRARY_PATH` to include the 32-bit Oracle libraries.

> **Note:**
>
> To view the libraries that are required by an Oracle GoldenGate process, use the `ldd` *process* shell command before starting the process. This command also shows an error message for any that are missing.

Example 2-1 Setting a library path

```
export LD_LIBRARY_PATH=/ggs/10.0:$LD_LIBRARY_PATH
```

2.4 Preparing to Install Oracle GoldenGate Within a Cluster

This topic covers the installation requirements that apply when Oracle GoldenGate will be installed in a cluster environment. Oracle GoldenGate can be used with any cluster-management solution that has the ability to automate failover.

2.4.1 Deciding Where to Install Oracle GoldenGate Binaries and Files in the Cluster

You will need to install at least some Oracle GoldenGate objects on shared storage. Select cluster-aware shared storage that is independent of, but available to, all nodes of the cluster.

The best practice is the install Oracle GoldenGate entirely on shared storage. This allows you to start the Oracle GoldenGate processes from any of the nodes without having to make changes to the parameter files. If the active node fails, the processes can be started quickly on another node, using the processing checkpoints that are preserved in the installation directory.

If you decide to install the Oracle GoldenGate binaries and files on each node, rather than on shared storage, the following must be true:

- The Oracle GoldenGate installation must have the same location path on every node
- At minimum, install the following directories on the shared storage to support Oracle GoldenGate recovery requirements. On UNIX or Linux, you can create symbolic links to them from the installation directory on each node.
  - `dirchk`
  - `dirdat`

  These directories are among those created when you issue `CREATE SUBDIRS` during installation.
- The parameter files in the `dirprm` directory, if not placed on the shared drive, must be identical on all nodes. To resolve environment settings that must be different from one node to the other, you can set environment settings so they are inherited from the local Manager process or reference a node-specific Oracle GoldenGate

macro file. Because this scenario can be difficult to enforce, the inherent concerns can be avoided by storing the parameter files on the shared drive.

See also Integrating Oracle GoldenGate into a Cluster after you install Oracle GoldenGate.

2.5 Installing Oracle GoldenGate on Linux and UNIX

Follow these steps to install Oracle GoldenGate for Oracle on a Linux or UNIX system or in the appropriate location in a cluster. See Preparing to Install Oracle GoldenGate Within a Cluster for more information.This procedure expands the Oracle GoldenGate installation files and creates the working subdirectories.

1. Extract the Oracle GoldenGate `mediapack.zip` file to the system and directory where you want Oracle GoldenGate to be installed.
2. Run the command shell.
3. Change directories to the new Oracle GoldenGate directory.
4. From the Oracle GoldenGate directory, run the GGSCI program.

   ```
   GGSCI
   ```
5. In GGSCI, issue the following command to create the Oracle GoldenGate working directories.

   ```
   CREATE SUBDIRS
   ```
6. Issue the following command to exit GGSCI.

   ```
   EXIT
   ```

2.6 Installing Oracle GoldenGate on Windows

Follow these steps to install Oracle GoldenGate for Oracle on a Windows system or in the appropriate location in a cluster. See Preparing to Install Oracle GoldenGate Within a Cluster for more information.

Installing Oracle GoldenGate on Windows requires the following steps:

Installing the Oracle GoldenGate Files

Specifying a Custom Manager Name

Installing Manager as a Windows Service

2.6.1 Installing the Oracle GoldenGate Files

These steps expand the installation files and create the working subdirectories in the Oracle GoldenGate installation directory.

1. (Windows Cluster) Log into one of the nodes in the cluster.
2. (Windows Cluster) Choose a drive for the Oracle GoldenGate installation location. This drive must be a resource within the same cluster group that contains the database instance.
3. (Windows Cluster) Ensure that this cluster group is owned by the cluster node that you are logging into.

4. Install Oracle GoldenGate according to the following instructions.
5. Unzip the downloaded file(s) by using WinZip or an equivalent compression product.
6. Move the files in binary mode to a folder on the drive where you want to install Oracle GoldenGate. *Do not* install Oracle GoldenGate into a folder that contains spaces in its name, even if the path is in quotes. For example:

   `C:\"Oracle GoldenGate"` is not valid.

   `C:\Oracle_GoldenGate` is valid.
7. From the Oracle GoldenGate folder, run the GGSCI program.
8. In GGSCI, issue the following command to create the Oracle GoldenGate working directories.

   ```
   CREATE SUBDIRS
   ```
9. Issue the following command to exit GGSCI.

   ```
   EXIT
   ```

2.6.2 Specifying a Custom Manager Name

You must specify a custom name for the Manager process if either of the following is true:

- You want to use a name for Manager other than the default of `GGSMGR`.
- There will be multiple Manager processes running as Windows services on this system. Each Manager on a system must have a unique name. Before proceeding further, note the names of any local Manager services.

1. From the directory that contains the Manager program, run GGSCI.
2. Issue the following command.

   ```
   EDIT PARAMS ./GLOBALS
   ```

   > **Note:**
   >
   > The `./` portion of this command must be used, because the `GLOBALS` file must reside at the root of the Oracle GoldenGate installation file.

3. In the file, add the following line, where *`name`* is a one-word name for the Manager service.

   `MGRSERVNAME` *`name`*
4. Save the file. The file is saved automatically with the name `GLOBALS`, without a file extension. Do not move this file. It is used during installation of the Windows service and during data processing.

2.6.3 Installing Manager as a Windows Service

By default, Manager is not installed as a service and can be run by a local or domain account. However, when run this way, Manager will stop when the user logs out.

When you install Manager as a service, you can operate it independently of user connections, and you can configure it to start manually or at system start-up.

Installing Manager as a service is required on a Windows Cluster, but optional otherwise.

1. (Recommended) Log on as the system administrator.
2. Click **Start** then **Run**, and type `cmd` in the **Run** dialog box.
3. From the directory that contains the Manager program that you are installing as a service, run the `install` program with the following syntax:

   ```
   install option [...]
   ```

   Where: *option* is one of the following:

Table 2-2 `INSTALL` options

Option	Description
`ADDEVENTS`	Adds Oracle GoldenGate events to the Windows Event Manager.
`ADDSERVICE`	Adds Manager as a service with the name that is specified with the `MGRSERVNAME` parameter in the `GLOBALS` file, if one exists, or by the default of `GGSMGR`. `ADDSERVICE` configures the service to run as the Local System account, the standard for most Windows applications because the service can be run independently of user logins and password changes. To run Manager as a specific account, use the `USER` and `PASSWORD` options.[1] The service is installed to start at system boot time (see `AUTOSTART`). To start it after installation, either reboot the system or start the service manually from the Services applet of the Control Panel.
`AUTOSTART`	Sets the service that is created with `ADDSERVICE` to start at system boot time. This is the default unless `MANUALSTART` is used.
`MANUALSTART`	Sets the service that is created with `ADDSERVICE` to start manually through GGSCI, a script, or the Services applet of the Control Panel. The default is `AUTOSTART`.
`USER` *name*	Specifies a domain user account that executes Manager. For *name*, include the domain name, a backward slash, and the user name, for example `HEADQT\GGSMGR`. By default, the Manager service is installed to use the Local System account.
`PASSWORD` *password*	Specifies the password for the user that is specified with `USER`.

[1] A user account can be changed by selecting the Properties action from the Services applet of the Windows Control Panel.

4. If Windows User Account Control (UAC) is enabled, you are prompted to allow or deny the program access to the computer. Select **Allow** to enable the `install` program to run. This installs the Manager service with a local system account running with administrator privileges. No further UAC prompts will be encountered when running Manager if installed as a service.

> **Note:**
>
> If Manager is not installed as a service, Oracle GoldenGate users will receive a UAC prompt to confirm the elevation of privileges for Manager when it is started from the GGSCI command prompt. Running other Oracle GoldenGate programs also triggers a prompt.

2.7 Setting up Oracle GoldenGate for MySQL Cluster

Oracle GoldenGate supports MySQL cluster in 11.2.1.0.4 and later versions and is certified on MySQL Cluster 7.1.15 and MySQL 5.1.*x*. The following procedures show how you can set up Oracle GoldenGate to work for MySQL cluster.

- Step 1: Install MySQL Cluster
- Step 2: Start MySQL Cluster
- Step 3: Install and Run Oracle GoldenGate

2.7.1 Step 1: Install MySQL Cluster

Install MySQL Cluster as described in the MySQL document "*Installing MySQL Cluster on Linux*" at:

`https://dev.mysql.com/doc/refman/5.7/en/mysql-cluster-install-linux.html`

> **Note:**
>
> Oracle GoldenGate now supports MySQL Cluster/NDB storage mode only. Therefore, the following setup is needed in the configuration file (i.e. `my.cnf`):
>
> ```
> default-storage-engine=ndbcluster
> ```
>
> After you create MySQL database and tables, you can check the status to make sure correct engine and data type is used. The following is an example:
>
> ```
> show table status where Name='te';
> ```

2.7.2 Step 2: Start MySQL Cluster

To use Oracle GoldenGate for MySQL Cluster, you need to disable the `NDB-LOG-UPDATE-AS-WRITE` and `NDB-LOG-UPDATED-ONLY` parameters inside their respective MySQL database configuration files (`my.cnf`/`my.ini`) or by using the `mysqld` command. Extract must be positioned to a point in time where the disabled options are already in effect; for example:

```
mysqld --default-file=/rdbms/mysql/myssqlcluster/my_cluster/conf/my.cnf -uroot --
datadir=/rdbms/mysql/myssqlcluster/my_cluster/mysqld_data--ndbcluster --ndb-log-
update-as-write=0 --ndb-log-updated-only=0 --default-storage-engine=ndbcluster --
max_allowed_packet=1G
```

The `--ndb-log-updated-only` option for MySQL does the following:

- Logs complete rows.
- Logs only column data that has been updated; that is, column data whose value has been set, regardless of whether or not the value was actually changed. This is the default behavior. If you need to log full rows, you can do so by setting `--ndb-log-updated-only` to 0 or `OFF`.

The `--ndb-log-update-as-write` option logs changed data as `UPDATE` operations.

2.7.3 Step 3: Install and Run Oracle GoldenGate

Follow the instructions in Installing Oracle GoldenGate on Linux and UNIX to install Oracle GoldenGate. To support the cluster environment, install Oracle GoldenGate on a shared file system that can be accessed by all of the cluster nodes.

2.8 Integrating Oracle GoldenGate into a Cluster

If you installed Oracle GoldenGate in a cluster, take the following steps to integrate Oracle GoldenGate within the cluster solution.

2.8.1 General Requirements in a Cluster

These instructions apply generically and may not be applicable to your specific cluster system.

You must meet the following requirements when integration Oracle GoldenGate into a cluster:

1. Register the Oracle GoldenGate Manager process (and only Manager) as a cluster-managed resource as you would any other application. Manager must be the only Oracle GoldenGate process that the cluster-management software starts and stops, because it is the parent process that manages all other processes.
2. If the cluster uses a virtual IP address, you may need to obtain an available fixed IP address for the Manager process. The VIP must be an available IP address on the public subnet and cannot be determined through DHCP. In the parameter files of the Extract data pumps, specify the VIP of the remote Manager as the input value of the `RMTHOST` parameter. Other Oracle GoldenGate products that access Manager also should use the VIP.
3. Make certain that all nodes in the cluster have synchronized system clocks. The clocks must be synchronized with the clock on the system where Extract is executed. Oracle GoldenGate compares the time of the local system to the commit timestamps to make critical decisions. For information about synchronizing system clocks, consult your systems administrator.
4. When you configure Manager, add the `AUTOSTART` and `AUTORESTART` parameters so that Manager starts the replication processes automatically. You can, when needed, control Extract, Replicat, and other Oracle GoldenGate processes from within the Oracle GoldenGate user interfaces.
5. Mount the shared drive on one node only. This prevents processes from being started on another node. Use the same mount point on all nodes.
6. Configure Oracle GoldenGate as appropriate for your environment.

2.8.2 Adding Oracle GoldenGate as a Windows Cluster Resource

When installing Oracle GoldenGate in a Windows cluster, follow these instructions to establish Oracle GoldenGate as a cluster resource and configure the Manager service correctly on all nodes.

- In the cluster administrator, add the Manager process to the group that contains the database instance to which Oracle GoldenGate will connect.
- Make sure all nodes on which Oracle GoldenGate will run are selected as possible owners of the resource.
- Make certain the Manager Windows service has the following dependencies (configurable from the Services control panel):
  - The database resource
  - The disk resource that contains the Oracle GoldenGate directory
  - The disk resource that contains the database transaction log files
  - The disk resource that contains the database transaction log backup files

3 Preparing the System for Oracle GoldenGate

This chapter contains instructions for preparing your system for running Oracle GoldenGate. It is comprised of the following sections:

3.1 Ensuring Data Availability

Retain enough binary log data so that if you stop Extract or there is an unplanned outage, Extract can start again from its checkpoints. Extract must have access to the binary log that contains the start of the oldest uncommitted unit of work, and all binary logs thereafter. The recommended retention period is at least 24 hours worth of transaction data, including both active and archived information. You might need to do some testing to determine the best retention time given your data volume and business requirements.

If data that Extract needs during processing was not retained, either in active or backup logs, one of the following corrective actions might be required:

- Alter Extract to capture from a later point in time for which binary log data is available (and accept possible data loss on the target).
- Resynchronize the source and target tables, and then start the Oracle GoldenGate environment over again.

To determine where the Extract checkpoints are, use the `INFO EXTRACT` command. For more information, see .

3.2 Setting Logging Parameters

To capture from the MySQL transaction logs, the Oracle GoldenGate Extract process must be able to find the index file. index file in turn contains the paths of all binary log files.

> **Note:**
>
> Extract expects that all of the table columns are in the binary log. As a result, only `binlog_row_image` set as `full` is supported and this is the default. Other values of `binlog_row_image` are not supported.

Extract checks the following parameter settings to get this index file path:

1. Extract `TRANLOGOPTIONS` parameter with the `ALTLOGDEST` option: If this parameter specifies a location for the log index file, Extract accepts this location over any default that is specified in the MySQL Server configuration file. When `ALTLOGDEST` is used, the binary log index file must also be stored in the specified directory. This parameter should be used if the MySQL configuration file does not specify the full index file path name, specifies an incorrect location, or if there are multiple installations of MySQL on the same machine

   To specify the index file path with `TRANLOGICOPTIONS` with `ALTLOGDEST`, use the following command format on Windows:

   ```
   TRANLOGOPTIONS ALTLOGDEST "C:\\Program Files\\MySQL\\logs\\binlog.index"
   ```

   On Linux, use this format:

   ```
   TRANLOGOPTIONS ALTLOGDEST "/mnt/rdbms/mysql/data/logs/binlog.index"
   ```

2. The MySQL Server configuration file: The configuration file stores default startup options for the MySQL server and clients. On Windows, the name of the configuration file is `my.ini`. On other platforms, it is `my.cnf`. In the absence of `TRANLOGOPTIONS` with `ALTLOGDEST`, Extract gets information about the location of the log files from the configuration file; however, even with `ALTLOGDEST`, these Extract parameters must be set correctly:

   - `binlog-ignore-db=oggddl`: This prevents DDL logging history table entries in the `binlog` and is set in the `my.cnf` or `my.ini` file.
   - `log-bin`: This parameter is used to enable binary logging. This parameter also specifies the location of the binary log index file and is a required parameter for Oracle GoldenGate, even if `ALTLOGDEST` is used. If log-bin is not specified, binary logging will be disabled and Extract returns an error.
   - `log-bin-index`: This parameter specifies the location of the binary log index. If it is not used, Extract assumes that the index file is in the same location as the log files. If this parameter is used and specifies a different directory from the one that contains the binary logs, the binary logs must not be moved once Extract is started.
   - `max_binlog_size`: This parameter specifies the size, in bytes, of the binary log file.

     > **Note:**
     >
     > The server creates a new binary log file automatically when the size of the current log reaches the `max_binlog_size` value, unless it must finish recording a transaction before rolling over to a new file.

- `binlog_format`: This parameter sets the format of the logs. It must be set to the value of `ROW`, which directs the database to log DML statements in binary format. Any other log format (`MIXED` or `STATEMENT`) causes Extract to abend.

> **Note:**
>
> MySQL binary logging does not allow logging to be enabled or disabled for specific tables. It applies globally to all tables in the database.

To locate the configuration file, Extract checks the `MYSQL_HOME` environment variable: If `MYSQL_HOME` is set, Extract uses the configuration file in the specified directory. If `MYSQL_HOME` is not set, Extract queries the `information_schema.global_variables` table to determine the MySQL installation directory. If a configuration file exists in that directory, Extract uses it.

3.3 Adding Host Names

Oracle GoldenGate gets the name of the database it is supposed to connect to from the `SOURCEDB` parameter. A successful connection depends on the localhost entry being properly configured in the system host file. To avoid issues that arise from improper local host configuration, you can use `SOURCEDB` in the following format:

`SOURCEDB` *`database_name@host_name`*

Where: *`database_name`* is the name of the MySQL instance, and *`host_name`* is the name or IP address of the local host. If using an unqualified host name, that name must be properly configured in the DNS database. Otherwise, use the fully qualified host name, for example `myhost.company.com`.

3.4 Setting the Session Character Set

The `GGSCI`, Extract and Replicat processes use a session character set when connecting to the database. For MySQL, the session character set is taken from the `SESSIONCHARSET` option of `SOURCEDB` and `TARGETDB`. Make certain you specify a session character set in one of these ways when you configure Oracle GoldenGate.

3.5 Configuring Bi-Directional Replication

In a bi-directional configuration, there are Extract and Replicat processes on both the source and target systems to support the replication of transactional changes on each system to the other system. To support this configuration, each Extract must be able to filter the transactions applied by the local Replicat, so that they are not recaptured and sent back to their source in a continuous loop. Additionally, `AUTO_INCREMENT` columns must be set so that there is no conflict between the values on each system.

1. Configure Oracle GoldenGate for high availability or active-active replication according to the instructions in the .
2. To filter out Replicat operations in a bi-directional configuration so that the applied operations are not captured and looped back to the source again, take the following steps on each MySQL database:

- Configure each Replicat process to use a checkpoint table. Replicat writes a checkpoint to this table at the end of each transaction. You can use one global checkpoint table or one per Replicat process See .
- Specify the name of the checkpoint table with the `FILTERTABLE` option of the `TRANLOGOPTIONS` parameter in the Extract parameter file. The Extract process will ignore transactions that end with an operation to the specified table, which should only be those of Replicat.

> **Note:**
>
> Although optional for other supported databases as a means of enhancing recovery, the use of a checkpoint table is required for MySQL when using bi-directional replication (and likewise, will enhance recovery).

3. Edit the MySQL server configuration file to set the `auto_increment_increment` and `auto_increment_offset` parameters to avoid discrepancies that could be caused by the bi-directional operations. The following illustrates these parameters, assuming two servers: **ServerA** and **ServerB**.

   ServerA:

   ```
   auto-increment-increment = 2
   auto-increment-offset = 1
   ```

   ServerB:

   ```
   auto-increment-increment = 2
   auto-increment-offset = 2
   ```

3.6 Other Oracle GoldenGate Parameters for MySQL

The following parameters may be of use in MySQL installations, and might be required if non-default settings are used for the MySQL database. Other Oracle GoldenGate parameters will be required in addition to these, depending on your intended business use and configuration.

Table 3-1 Other Parameters for Oracle GoldenGate for MySQL

Parameter	Description
`DBOPTIONS` with `CONNECTIONPORT` *`port_number`*	Required to specify to the VAM the TCP/IP connection port number of the MySQL instance to which an Oracle GoldenGate process must connect if MySQL is not running on the default of 3306. `DBOPTIONS CONNECTIONPORT 3307`
`DBOPTIONS` with `HOST` *`host_id`*	Specifies the DNS name or IP address of the system hosting MySQL to which Replicat must connect.
`DBOPTIONS` with `ALLOWLOBDATATRUNCATE`	Prevents Replicat from abending when replicated LOB data is too large for a target MySQL `CHAR`, `VARCHAR`, `BINARY` or `VARBINARY` column.

Table 3-1 (Cont.) Other Parameters for Oracle GoldenGate for MySQL

Parameter	Description

`SOURCEDB with USERID and PASSWORD`	Specifies database connection information consisting of the database, user name and password to use by an Oracle GoldenGate process that connects to a MySQL database. If MySQL is not running on the default port of 3306, you must specify a complete connection string that includes the port number: `SOURCEDB` *`dbname@hostname:port`*, `USERID` *`user`*, `PASSWORD` *`password`*.Example:
	`SOURCEDB mydb@mymachine:3307, USERID myuser, PASSWORD mypassword`
	If you are not running the MySQL database on port 3306, you must also specify the connection port of the MySQL database in the `DBLOGIN` command when issuing commands that affect the database through GGSCI:
	`DBLOGIN SOURCEDB` *`dbname@hostname:port`*`, USERID` *`user`*`, PASSWORD` *`password`*
	For example:
	`GGSCI> DBLOGIN SOURCEDB mydb@mymachine:3307, USERID myuser, PASSWORD mypassword`

`SQLEXEC`	To enable Replicat to bypass the MySQL connection timeout, configure the following command in a `SQLEXEC` statement in the Replicat parameter file. `SQLEXEC "select CURRENT_TIME();" EVERY` *`n`* `MINUTES` **Where**: *`n`* is the maximum interval after which you want Replicat to reconnect. The recommended connection timeout 31536000 seconds (365 days).

See .

See .

3.7 Preparing Tables for Processing

This section describes how to prepare the tables for processing. Table preparation requires these tasks:

3.7.1 Assigning Row Identifiers

Oracle GoldenGate requires some form of unique row identifier on the source and target tables to locate the correct target rows for replicated updates and deletes.

3.7.1.1 How Oracle GoldenGate Determines the Kind of Row Identifier to Use

Unless a `KEYCOLS` clause is used in the `TABLE` or `MAP` statement, Oracle GoldenGate selects a row identifier to use in the following order of priority:

1. Primary key

2. First unique key alphanumerically that does not contain a timestamp or non-materialized computed column
3. If none of the preceding key types exist (even though there might be other types of keys defined on the table) Oracle GoldenGate constructs a pseudo key of all columns that the database allows to be used in a unique key, excluding those that are not supported by Oracle GoldenGate in a key or those that are excluded from the Oracle GoldenGate configuration.

> **Note:**
>
> If there are other, non-usable keys on a table or if there are no keys at all on the table, Oracle GoldenGate logs an appropriate message to the report file. Constructing a key from all of the columns impedes the performance of Oracle GoldenGate on the source system. On the target, this key causes Replicat to use a larger, less efficient `WHERE` clause.

3.7.1.2 Tables with a Primary Key Derived from a Unique Index

In the absence of a primary key on a table, MySQL will promote a unique index to primary key if the indexed column is `NOT NULL`. If there are more than one of these not-null indexes, the first one that was created becomes the primary key. To avoid Replicat errors, create these indexes in the same order on the source and target tables.

For example, assume that source and target tables named `ggvam.emp` each have columns named first, middle, and last, and all are defined as `NOT NULL`. If you create unique indexes in the following order, Oracle GoldenGate will abend on the target because the table definitions do not match.

Source:

```
mysql> create unique index uq1 on ggvam.emp(first);
mysql> create unique index uq2 on ggvam.emp(middle);
mysql> create unique index uq3 on ggvam.emp(last);
```

Target:

```
mysql> create unique index uq1 on ggvam.emp(last);
mysql> create unique index uq2 on ggvam.emp(first);
mysql> create unique index uq3 on ggvam.emp(middle);
```

The result of this sequence is that MySQL promotes the index on the source "first" column to primary key, and it promotes the index on the target "last" column to primary key. Oracle GoldenGate will select the primary keys as identifiers when it builds its metadata record, and the metadata will not match. To avoid this error, decide which column you want to promote to primary key, and create that index first on the source and target.

3.7.1.3 How to Specify Your Own Key for Oracle GoldenGate to Use

If a table does not have one of the preceding types of row identifiers, or if you prefer those identifiers not to be used, you can define a substitute key if the table has columns that always contain unique values. You define this substitute key by including a `KEYCOLS` clause within the Extract `TABLE` parameter and the Replicat `MAP` parameter.

The specified key will override any existing primary or unique key that Oracle GoldenGate finds.

3.7.2 Limiting Row Changes in Tables That Do Not Have a Key

If a target table does not have a primary key or a unique key, duplicate rows can exist. In this case, Oracle GoldenGate could update or delete too many target rows, causing the source and target data to go out of synchronization without error messages to alert you. To limit the number of rows that are updated, use the `DBOPTIONS` parameter with the `LIMITROWS` option in the Replicat parameter file. `LIMITROWS` can increase the performance of Oracle GoldenGate on the target system because only one row is processed.

3.7.3 Disabling Triggers and Cascade Constraints

Disable triggers, cascade delete constraints, and cascade update constraints on the target tables, or alter them to ignore changes made by the Oracle GoldenGate database user. Oracle GoldenGate replicates DML that results from a trigger or cascade constraint. If the same trigger or constraint gets activated on the target table, it becomes redundant because of the replicated version, and the database returns an error. Consider the following example, where the source tables are `emp_src` and `salary_src` and the target tables are `emp_targ` and `salary_targ`.

1. A delete is issued for `emp_src`.
2. It cascades a delete to `salary_src`.
3. Oracle GoldenGate sends both deletes to the target.
4. The parent delete arrives first and is applied to `emp_targ`.
5. The parent delete cascades a delete to `salary_targ`.
6. The cascaded delete from `salary_src` is applied to `salary_targ`.
7. The row cannot be located because it was already deleted in step 5.

3.8 Positioning Extract to a Specific Start Point

You can position the `ADD EXTRACT` and `ALTER EXTRACT` commands to a specific start point in the transaction logs with the following command.

`{ADD | ALTER EXTRACT}` *`group`*`, VAM, LOGNUM` *`log_num`*`, LOGPOS` *`log_pos`*

- *`group`* is the name of the Oracle GoldenGate Extract group for which the start position is required.
- *`log_num`* is the log file number. For example, if the required log file name is `test.000034`, this value is 34. Extract will search for this log file.
- *`log_pos`* is an event offset value within the log file that identifies a specific transaction record. Event offset values are stored in the header section of a log record. To position at the beginning of a `binlog` file, set the `log_pos` as 4. The `log_pos` 0 or 1 are not valid offsets to start reading and processing.

In MySQL logs, an event offset value can be unique only within a given binary file. The combination of the position value and a log number will uniquely identify a transaction record and cannot exceed a length of 37. Transactional records available after this

position within the specified log will be captured by Extract. In addition, you can position an Extract using a timestamp.

3.9 Changing the Log-Bin Location

Modifying the binary log location by using the `log-bin` variable in the MySQL configuration file might result in two different path entries inside the index file, which could result in errors. To avoid any potential errors, change the log-bin location by doing the following:

1. Stop any new DML operations.
2. Let the extract finish processing all of the existing binary logs. You can verify this by noting when the checkpoint position reaches the offset of the last log.
3. After Extract finishes processing the data, stop the Extract group and, if necessary, back up the binary logs.
4. Stop the MySQL database.
5. Modify the `log-bin` path for the new location.
6. Start the MySQL database.
7. To clean the old log name entries from index file, use `flush master` or `reset master` (based on your MySQL version).
8. Start Extract.

4
Uninstalling Oracle GoldenGate

This chapter contains procedures for uninstalling Oracle GoldenGate for My SQL. It contains the following sections:

- Uninstalling Oracle GoldenGate from Linux or UNIX
- Removing Oracle GoldenGate from Windows Cluster
- Uninstalling Oracle GoldenGate from Windows (Non-Cluster)

This procedure assumes that you no longer need the data in the Oracle GoldenGate trails, and that you no longer need to preserve the current Oracle GoldenGate environment. To preserve your current environment and data, make a backup of the Oracle GoldenGate directory and all subdirectories before starting this procedure.

4.1 Uninstalling Oracle GoldenGate from Linux or UNIX

To uninstall Oracle GoldenGate from Linux or UNIX machines, use this procedure:

1. Run the command shell.
2. (Suggested) Log on as the system administrator or as a user with permission to issue Oracle GoldenGate commands and delete files and directories from the operating system.
3. Change directories to the Oracle GoldenGate installation directory.
4. Run GGSCI.
5. Stop all Oracle GoldenGate processes.
6. Stop the Manager process.
7. (On a source system, optional) Disable binary logging.
8. (On any system where a Replicat checkpoint table is being used) Log into the database with the `DBLOGIN` command, and then remove the Replicat checkpoint table by running the `DELETE CHECKPOINTTABLE` command.
9. On all systems:
    - Make certain all Oracle GoldenGate processes are stopped (including GGSCI).
    - Remove the Oracle GoldenGate files by removing the installation directory.

4.2 Removing Oracle GoldenGate from Windows Cluster

To uninstall Oracle GoldenGate from Windows Cluster machines, use this procedure:

1. Working from the node in the cluster that owns the cluster group that contains the Manager resource, run GGSCI and then stop any Extract and Replicat processes that are still running.
2. Use the Cluster Administrator tool to take the Manager resource offline.

3. Right click the resource and select **Delete** to remove it.
4. Click **Start > Run**, and type cmd in the Run dialog box to open the command console.
5. Change directories to the Oracle GoldenGate installation directory.
6. Run the install program using the following syntax.

   ```
   install deleteevents deleteservice
   ```

   This command stops Oracle GoldenGate events from being reported to the Windows Event Manager and removes the Manager service.
7. Move the cluster group to the next node in the cluster, and repeat from 4.
8. Follow the instructions in "Uninstalling Oracle GoldenGate from Windows (Non-Cluster)".

4.3 Uninstalling Oracle GoldenGate from Windows (Non-Cluster)

To uninstall Oracle GoldenGate from non-cluster Windows machines, use this procedure:

1. (Recommended) Log on as the system administrator or as a user with permission to issue Oracle GoldenGate commands and to delete files and directories from the operating system.
2. From the Oracle GoldenGate installation folder, run GGSCI.
3. Stop all Oracle GoldenGate processes.
4. Stop the Manager program or service.

> **Note:**
>
> Skip step 5 through step 8 if you already performed them when removing Oracle GoldenGate from a Windows cluster.

5. Click **Start** then **Run**, and type `cmd` in the **Run** dialog box to open the command console.
6. Change directories to the Oracle GoldenGate installation directory.
7. Run the `install` program using the following syntax.

   ```
   install deleteevents deleteservice
   ```

   This command stops Oracle GoldenGate events from being reported to the Windows Event Manager and removes the Manager service.
8. (On a source system, optional) Disable binary logging.
9. (On any system where a Replicat checkpoint table is being used) Log into the database with the `DBLOGIN` command, and then remove the Replicat checkpoint table by running the `DELETE CHECKPOINTTABLE` command.
10. On all systems:

Make certain all processes are stopped (including GGSCI) and then remove the Oracle GoldenGate files by removing the installation directory.

A

Oracle GoldenGate Installed Components

This appendix describes the programs, directories, and other components created or used by the Oracle GoldenGate software in the Oracle GoldenGate installation directory. Additional files not listed here might be installed on certain platforms. Files listed here might not be installed on every platform.
This appendix contains the following sections:

A.1 Oracle GoldenGate Programs and Utilities

This section describes programs installed in the root Oracle GoldenGate installation directory.

> **Note:**
>
> Some programs may not exist in all installations. For example, if only capture or delivery is supported by Oracle GoldenGate for your platform, the extract or replicat program will not be installed, respectively. Likewise, special files might be installed to support a specific database.

Table A-1 Oracle GoldenGate Installed Programs and Utilities

Program	Description
`convchk`	Converts checkpoint files to a newer version.
`convprm`	Converts parameter files that do not use SQL-92 rules for quoted names and literals to updated parameter files that use SQL-92 rules. SQL-92 format for quoted object names and literals was introduced as the default with version 12c of Oracle GoldenGate.
`defgen`	Generates data definitions and is referenced by Oracle GoldenGate processes when source and target tables have dissimilar definitions.
`emsclnt`	Sends event messages created by Collector and Replicat on Windows or UNIX systems to EMS on NonStop systems.
`extract`	Performs capture from database tables or transaction logs or receives transaction data from a vendor access module.
`ggmxinstall`	Oracle GoldenGate installation script for the SQL/MX database.

Table A-1 (Cont.) Oracle GoldenGate Installed Programs and Utilities

Program	Description
ggcmd	Associated program of ggsci. Launches and monitors external applications, such as the JAGENT of Oracle GoldenGate Monitor. Integrates those applications into the ggsci environment.
ggsci	User interface to Oracle GoldenGate for issuing commands and managing parameter files.
ggsmgr.jcl ggsmgr.proc ggsmgrst.jcl ggsmgrst.proc	Start the Oracle GoldenGate Manager process from a batch job or the operator console on a z/OS system. Installed to support DB2 z/OS databases.
install	Installs Oracle GoldenGate as a Windows service and provides other Windows-based service options.
keygen	Generates data-encryption keys.
logdump	A utility for viewing and saving information stored in extract trails or files.
mgr	(Manager) Control process for resource management, control and monitoring of Oracle GoldenGate processes, reporting, and routing of requests through the GGSCI interface.
oggerr	Manages Oracle GoldenGate error messages.
replicat	Applies data to target database tables.
reverse	A utility that reverses the order of transactional operations, so that Replicat can be used to back out changes from target tables, restoring them to a previous state.
server	The Collector process, an Extract TCP/IP server collector that writes data to remote trails.
vamserv	Started by Extract to read the TMF audit trails generated by TMF-enabled applications. Installed to support the NonStop SQL/MX database.

A.2 Oracle GoldenGate Subdirectories

This Section describes the subdirectories of the Oracle GoldenGate installation directory and their contents.

> **Note:**
>
> Some directories may not exist in all installations.

Table A-2 Oracle GoldenGate Installed Subdirectories

Directory	Description
br	Contains the checkpoint files for the bounded recover feature.

Table A-2 (Cont.) Oracle GoldenGate Installed Subdirectories

Directory	Description
`cfg`	Contains the property and XML files that are used to configure Oracle GoldenGate Monitor.
`dirdb`	Contains the data store that is used to persist information that is gathered from an Oracle GoldenGate instance for use by the Oracle GoldenGate Monitor application or within Oracle Enterprise Manager.

`dirchk`	Contains the checkpoint files created by Extract and Replicat processes, which store current read and write positions to support data accuracy and fault tolerance. Written in internal Oracle GoldenGate format.
	File name format is *group_name*+*sequence_number*.*ext* where *sequence_number* is a sequential number appended to aged files and *ext* is either `cpe` for Extract checkpoint files or `cpr` for Replicat checkpoint files.
	Do not edit these files.
	Examples:
	`ext1.cpe`
	`rep1.cpr`

`dircrd`	Contains credential store files.

`dirdat`	The default location for Oracle GoldenGate trail files and extract files that are created by Extract processes to store extracted data for further processing by the Replicat process or another application or utility. Written in internal Oracle GoldenGate format.
	File name format is a user-defined two-character prefix followed by either a six-digit sequence number (trail files) or the user-defined name of the associated Extract process group (extract files).
	Do not edit these files.
	Examples:
	`rt000001`
	`finance`

`dirdef`	The default location for data definitions files created by the `DEFGEN` utility to contain source or target data definitions used in a heterogeneous synchronization environment. Written in external ASCII. File name format is a user-defined name specified in the `DEFGEN` parameter file. These files may be edited to add definitions for newly created tables. If you are unsure of how to edit a definitions file, contact Oracle GoldenGate technical support. Example: `defs.dat`
`dirdmp`	Contains trace, or dump, files that support the internal activity logging mechanism.
`dirjar`	Contains the Java executable files that support Oracle GoldenGate Monitor.

Table A-2 (Cont.) Oracle GoldenGate Installed Subdirectories

Directory	Description

`dirpcs`	Default location for status files. File name format is *`group.extension`* where *`group`* is the name of the group and *`extension`* is either `pce` (Extract), `pcr` (Replicat), or `pcm` (Manager).
	These files are only created while a process is running. The file shows the program name, the process name, the port number, and the process ID.
	Do not edit these files.
	Examples:
	`mgr.pcm`
	`ext.pce`

`dirprm`	The default location for Oracle GoldenGate parameter files created by Oracle GoldenGate users to store run-time parameters for Oracle GoldenGate process groups or utilities. Written in external ASCII format. File name format is *`group name/user-defined name.prm`* or `mgr.prm` .
	These files may be edited to change Oracle GoldenGate parameter values after stopping the process. They can be edited directly from a text editor or by using the `EDIT PARAMS` command in GGSCI.
	Examples:
	`defgen.prm`
	`finance.prm`

`dirrec`	Not used by Oracle GoldenGate.

`dirrpt`	The default location for process report files created by Extract, Replicat, and Manager processes to report statistical information relating to a processing run. Written in external ASCII format.
	File name format is *`group name+sequence number.rpt`* where `sequence number` is a sequential number appended to aged files.
	Do not edit these files.
	Examples:
	`fin2.rpt`
	`mgr4.rpt`

`dirsql`	Used by the `triggen` utility to store SQL scripts before `triggen` was deprecated. Currently used to store training scripts and any user-created SQL scripts that support Oracle GoldenGate.
`dirtmp`	The default location for storing transaction data when the size exceeds the memory size that is allocated for the cache manager. Do not edit these files.
`dirwlt`	Contains Oracle GoldenGate wallet files.
`UserExitExamples`	Contains sample files to help with the creation of user exits.

A.3 Other Oracle GoldenGate Files

This section describes other files, templates, and objects created or installed in the root Oracle GoldenGate installation directory.

> **Note:**
>
> Some files may not be installed in your environment, depending on the database and OS platform.

Table A-3 Other Oracle GoldenGate Installed Files

Component	Description
`bcpfmt.tpl`	Template for use with Replicat when creating a run file for the Microsoft BCP/DTS bulk-load utility.
`bcrypt.txt`	Blowfish encryption software license agreement.
`cagent.dll`	Contains the Windows dynamic link library for the Oracle GoldenGate Monitor C sub-agent.
`category.dll`	Windows dynamic link library used by the `INSTALL` utility.
`chkpt_`*`db`*`_create.sql`	Script that creates a checkpoint table in the local database. A different script is installed for each database type.
`db2cntl.tpl`	Template for use with Replicat when creating a control file for the IBM `LOADUTIL` bulk-load utility.
`ddl_cleartrace.sql`	Script that removes the DDL trace file. (Oracle installations)
`ddl_ddl2file.sql`	Script that saves DDL from the marker table to a file.
`ddl_disable.sql`	Script that disables the Oracle GoldenGate DDL trigger. (Oracle installations)
`ddl_enable.sql`	Script that enables the Oracle GoldenGate DDL trigger. (Oracle installations)
`ddl_filter.sql`	Script that supports filtering of DDL by Oracle GoldenGate. This script runs programmatically; do not run it manually.
`ddl_nopurgeRecyclebin.sql`	Empty script file for use by Oracle GoldenGate support staff.
`ddl_ora11.sql` `ddl_ora12.sql`	Scripts that run programmatically as part of Oracle GoldenGate DDL support; do not run these scripts.
`ddl_pin.sql`	Script that pins DDL tracing, the DDL package, and the DDL trigger for performance improvements. (Oracle installations)
`ddl_purgeRecyclebin.sql`	Script that purges the Oracle recyclebin in support of the DDL replication feature.
`ddl_remove.sql`	Script that removes the DDL extraction trigger and package. (Oracle installations)
`ddl_session.sql` ddl_session1.sql	Supports the installation of the Oracle DDL objects. This script runs programmatically; do not run it manually.
`ddl_setup.sql`	Script that installs the Oracle GoldenGate DDL extraction and replication objects. (Oracle installations)
`ddl_status.sql`	Script that verifies whether or not each object created by the Oracle GoldenGate DDL support feature exists and is functioning properly. (Oracle installations)

Table A-3 (Cont.) Other Oracle GoldenGate Installed Files

Component	Description
`ddl_staymetadata_off.sql` `ddl_staymetadata_on.sql`	Scripts that control whether the Oracle DDL trigger collects metadata. This script runs programmatically; do not run it manually.
`ddl_trace_off.sql` `ddl_trace_on.sql`	Scripts that control whether DDL tracing is on or off.
`ddl_tracelevel.sql`	Script that sets the level of tracing for the DDL support feature. (Oracle installations)
`debug files`	Debug text files that may be present if tracing was turned on.
`demo_`*`db_scriptname`*`.sql` `demo_`*`more_db_scriptname`*`.sql`	Scripts that create and populate demonstration tables for use with tutorials and basic testing.
`.dmp files`	Dump files created by Oracle GoldenGate processes for tracing purposes.
`ENCKEYS`	User-created file that stores encryption keys. Written in external ASCII format.
`exitdemo.c`	User exit example.
`exitdemo_utf16.c`	User exit example that demonstrates how to use UTF16 encoded data in the callback structures for information exchanged between the user exit and the process.
`freeBSD.txt`	License agreement for FreeBSD.
`ggmessage.dat`	Data file that contains error, informational, and warning messages that are returned by the Oracle GoldenGate processes. The version of this file is checked upon process startup and must be identical to that of the process in order for the process to operate.
`ggserr.log`	File that logs processing events, messages, errors, and warnings generated by Oracle GoldenGate.
`ggsmsg.dll`	Windows dynamic link library used by the `install` program.
`GLOBALS`	User-created file that stores parameters applying to the Oracle GoldenGate instance as a whole.
`help.txt`	Help file for the GGSCI command interface.

`icudtxx.dll`
`icuinxx.dll`
`icuucxx.dll`
	Windows shared libraries for International Components for Unicode, where *xx* is the currently used version.

`jagent.bat`	Windows batch file for the Java Agent for Oracle GoldenGate Monitor.
`jagent.log` `jagentjni.log`	Log files for the Oracle GoldenGate Monitor Agent.
`jagent.sh`	UNIX shell script for the Java Agent for Oracle GoldenGate Monitor
`LGPL.txt`	Lesser General Public License statement. Applies to free libraries from the Free Software Foundation.

Table A-3 (Cont.) Other Oracle GoldenGate Installed Files

Component	Description
`libodbc.so`	ODBC file for Ingres 2.6 on Unix.
`libodbc.txt`	License agreement for `libodbc.so`.
`libxml2.dll`	Windows dynamic link library containing the XML library for the Oracle GoldenGate XML procedures.
`libxml2.txt`	License agreement for `libxml2.dll` .
`marker.hist`	File created by Replicat if markers were passed from a NonStop source system.
`marker_remove.sql`	Script that removes the DDL marker table. (Oracle installations)
`marker_setup.sql`	Script that installs the Oracle GoldenGate DDL marker table. (Oracle installations)
`marker_status.sql`	Script that confirms successful installation of the DDL marker table. (Oracle installations)
`notices.txt`	Third-party software license file.
`odbcinst.ini`	Ingres 2.6 on Unix ODBC configuration file.
`params.sql`	Script that contains configurable parameters for DDL support. (Oracle installations)
`pthread-win32.txt`	License agreement for `pthread-VC.dll` .
`pthread-VC.dll`	POSIX threads library for Microsoft Windows.
`prvtclkm.plb`	Supports the replication of Oracle encrypted data.
`pw_agent_util.bat` `pw_agent_util.sh`	Script files that support the Oracle GoldenGate Monitor Agent.
`role_setup.sql`	Script that creates the database role necessary for Oracle GoldenGate DDL support. (Oracle installations)
`sampleodbc.ini`	Sample ODBC file for Ingres 2.6 on UNIX.
`sqlldr.tpl`	Template for use with Replicat when creating a control file for the Oracle SQL*Loader bulk-load utility.
`start.prm` `stop.prm`	z/OS `paramlib` members to start and stop the Manager process.
`startmgr` `stopmgr`	z/OS Unix System Services scripts to start the Manager process from GGSCI.
`startmgrcom` `stopmgrcom`	z/OS system input command for the Manager process.
`tcperrs`	File containing user-defined instructions for responding to TCP/IP errors.
`usrdecs.h`	Include file for user exit API.
`xerces-c_2_8.dll`	Apache XML parser library.
`zlib.txt`	License agreement for `zlib` compression library.

A.4 Oracle GoldenGate Checkpoint Table

When database checkpoints are being used, Oracle GoldenGate creates a checkpoint table with a user-defined name in the database upon execution of the `ADD CHECKPOINTTABLE` command, or a user can create the table by using the `chkpt_db_create.sql` script (where *db* is an abbreviation of the type of database that the script supports). For a description of this table, see *Administering Oracle GoldenGate for Windows and UNIX*.