Oracle® GoldenGate
DB2 LUW Installation and Setup Guide
11*g* Release 2 Patch Set 1 (11.2.1.0.1)
E29798-01

April 2012

ORACLE®

Oracle GoldenGate DB2 LUWInstallation and Setup Guide 11g Release 2 (11.2.1.0.0)

E29798-01

Contents

. . . .

. . . .

. . . .

CHAPTER 1

System requirements and preinstallation instructions

. . . .

This chapter contains the requirements for the system and database resources that support Oracle GoldenGate.

Overview of Oracle GoldenGate for DB2 LUW

With Oracle GoldenGate for DB2 LUW, you can replicate data to and from similar or dissimilar supported DB2 LUW versions, or you can move data between a DB2 LUW database and a database of another type. Oracle GoldenGate for DB2 LUW supports the filtering, mapping, and transformation of data unless otherwise noted in this documentation.

Supported Platforms

To find out which Oracle GoldenGate builds are available for a specific combination of database version and operating system, log onto http://support.oracle.com and select the **Certifications** tab. For assistance, click **Tips for Finding Certifications.**

An e-mail and password are required to enter this site.

Operating system requirements

Memory requirements

The amount of memory that is required for Oracle GoldenGate depends on the number of concurrent processes that will be running. At minimum on the source system, there is a primary Extract process that captures source data and a secondary Extract data-pump process that transfers data across the network. At minimum on the target system is at least one Replicat process that applies the replicated data to the target database. In some cases, these processes might all operate on the same system, depending on the required configuration.

It is possible that you will need to use additional, parallel processes to improve throughput if your environment generates a large volume of transactional data that must be replicated. Oracle GoldenGate supports up to 5,000 concurrent Extract and Replicat processes per instance of Oracle GoldenGate. Each Extract and Replicat process needs approximately 25-55 MB of memory, or more depending on the size of the transactions and the number of concurrent transactions.

The actual amount of physical memory that is used by any Oracle GoldenGate process is controlled by the operating system, not the Oracle GoldenGate program. The Oracle GoldenGate cache manager takes advantage of the memory management functions of the

. . . .

operating system to ensure that Oracle GoldenGate processes work in a sustained and efficient manner. For more information about evaluating Oracle GoldenGate memory requirements, see the CACHEMGR parameter in the Oracle GoldenGate *Windows and UNIX Reference Guide*.

Disk requirements

Assign the following free disk space:

- 50-150 MB, depending on the database and platform. This includes space for the compressed download file and space for the uncompressed files. You can delete the download file after the installation is complete.
- 40 MB for the working directories and binaries for each instance of Oracle GoldenGate that you are installing on the system. For example, to install two builds of Oracle GoldenGate into two separate directories, allocate 80 MB of space.
- 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 the Oracle GoldenGate *Administration Guide*.

Storage for Oracle GoldenGate trails

To prevent trail activity from interfering with business applications, assign a separate disk or file system to contain the trail files. These files are created during processing to store all of the data that is captured by Oracle GoldenGate. The default size is 10 megabytes, but can be changed during the configuration process. Trail files accumulate but can be purged according to rules set with the PURGEOLDEXTRACTS parameter.

Trail files can reside on drives that are local to the Oracle GoldenGate installation, or they can reside on NAS or SAN devices. You will specify the location of the trails when you configure Oracle GoldenGate.

For trails that are stored at the source location, there should be enough space to handle data accumulation should the network connection fail. In a typical configuration, a secondary Extract process (known as a data pump) sends data from a local trail over the network, and it will fail when the network does. However, the primary Extract that reads the transaction logs and writes to the local trail will continue to do so. This Extract should not be stopped during a failure; otherwise, transaction data might be missed if the transaction logs recycle or get removed from the system before the data is completely captured. There must be enough disk space to hold the data accumulation.

For trails at the target location, provide enough disk space to handle data accumulation according to the purge rules set with the PURGEOLDEXTRACTS parameter. Even with PURGEOLDEXTRACTS in use, data will always accumulate on the target because it is transferred across the network faster than it can be applied to the target database.

To estimate required trail space

1. Estimate the longest time that the network could be unavailable. Plan to store enough data to withstand the longest possible outage, because otherwise you will need to resynchronize the source and target data if the outage outlasts disk capacity.

2. Estimate how much transaction log volume your business applications generate in one hour.

3. Use the following formula to calculate the required disk space.

[log volume in one hour] x [number of hours downtime] x .4 = trail disk space

This equation uses a multiplier of 40 percent because only about 40 percent of the data in a transaction log is needed by Oracle GoldenGate.

> NOTE   This formula is a conservative estimate, and you should run tests once you have configured Oracle GoldenGate to determine exactly how much space you need.

Network

- 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 the Oracle GoldenGate *Windows and UNIX Administrator's Guide* 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.

Operating system privileges

- 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.

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.

Other programs

- Before installing Oracle GoldenGate on a Windows system, install and configure the Microsoft Visual C ++ 2005 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.

Database requirements

Database configuration

- The Oracle GoldenGate Extract process calls the DB2READLOG function in the Administrative API to read the transaction log files of a DB2 LUW source database. In addition to DB2READLOG, Extract uses a small number of other API routines to check the source database configuration on startup.

- The Oracle GoldenGate Replicat process uses the DB2 CLI interface on a DB2 LUW target database. For instructions on installing this interface, see the DB2 documentation.

Database user for Oracle GoldenGate processes

- 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.

- Assign system administrator (SYSADM) or database administrator (DBADM) authority to the database user under which Extract runs. To give the Extract user DBADM authority, a user with SYSADM authority can issue the following grant statement.

```
GRANT DBADM ON DATABASE TO USER <user>
```

This authority can also be granted from the **User and Group Objects** folder in the DB2 Control Center. The database tab for the user that is assigned to an Oracle GoldenGate process should have the Database Administrative Authority box checked.

NOTE   If the Extract user does not have the required authority, Extract will log the following
errors and stop.

```
[SC=-1224:SQL1224N A database agent could not be started to
service a request, or was terminated as a result of a database
system shutdown or a force command.
SQL STATE 55032: The CONNECT statement is invalid, because the
database manager was stopped after this application was
started]
```

● Grant at least the following privileges to the database user under which Replicat runs:
  ❍ Local CONNECT to the target database
  ❍ SELECT on the system catalog views
  ❍ SELECT, INSERT, UPDATE, and DELETE on the target tables

Supported DB2 LUW data types

Oracle GoldenGate supports all DB2 LUW data types, except those listed in "Non-supported DB2 LUW data types".

Limitations of support

● Oracle GoldenGate supports multi-byte character data types and multi-byte data stored in character columns. Multi-byte data is only supported in a like-to-like configuration. Transformation, filtering, and other types of manipulation are not supported for multi-byte character data.

● BLOB and CLOB columns must have a LOGGED clause in their definitions.

● GRAPHIC and VARGRAPHIC columns must be in a database where the character set is UTF16. Any other character set causes the Oracle GoldenGate to abend.

● 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.

● Extract fully supports the capture and apply of TIMESTAMP(0) through TIMESTAMP(6). Extract also captures TIMESTAMP(7) through TIMESTAMP(12), but it truncates the data to microseconds (maximum of six digits of fractional time) and issues a warning to the error log. Replicat truncates timestamp data from other sources to microseconds when applying it to TIMESTAMP(7) through TIMESTAMP(12) in a DB2 LUW target.

● When the size of a large object exceeds 4K, Oracle GoldenGate stores the data in segments within the Oracle GoldenGate trail. The first 4K is stored in the base segment, and the rest is stored in a series of 2K segments. Oracle GoldenGate does not support the filtering, column mapping, or manipulation of large objects of this size. Full Oracle GoldenGate functionality can be used for objects that are 4K or smaller.

Non-supported DB2 LUW data types

- XMLType
- DECFLOAT
- User-defined types
- Negative dates

Supported objects and operations for DB2 LUW

- Oracle GoldenGate supports the maximum number of columns and column size per table that is supported by the database.
- TRUNCATE TABLE for DB2 LUW version 9.7 and later.
- Multi Dimensional Clustered Tables (MDC) for DB2 LUW 9.5 and later.
- Materialized Query Tables. Oracle GoldenGate does not replicate the MQT itself, but only the base tables. The target database automatically maintains the content of the MQT based on the changes that are applied to the base tables by Replicat.
- Extraction and replication of compressed data (CREATE TABLESPACE COMPRESS YES).
- Tables with ROW COMPRESSION in DB2 LUW versions 9.7 and later.
- Generated columns (GENERATE ALWAYS clause)

Non-supported objects and operations for DB2 LUW

- Extraction and replication of DML operations on DB2 tables that contain rows of up to 512 KB in length. This size exceeds the maximum row size of DB2.
- Schema, table or column names that have trailing spaces.
- Multiple instances of a database
- Datalinks
- Extraction or replication of DDL (data definition language) operations
- Tables with ROW COMPRESSION in DB2 LUW versions 9.5 and earlier.
- Tables with VALUE COMPRESSION (all versions).

  NOTE To include tables with non-supported types of table compression in the Oracle GoldenGate configuration, deactivate the non-supported compression and then reorganize the tables; otherwise, exclude them from the Oracle GoldenGate configuration.

Supported and non-supported object names and case

The following will help you verify whether the name of a supported object qualifies or disqualifies it for inclusion in an Oracle GoldenGate configuration.

Object names and owners

Source and target object names must be fully qualified in Oracle GoldenGate parameter files, as in fin.emp. Oracle GoldenGate supports character case as follows.

Case sensitivity

The following are general guidelines for the case-sensitivity of object names as it relates to Oracle GoldenGate. These guidelines may or may not apply to your databases, depending on whether the database or the underlying operating system supports case-sensitivity. Keep in mind that case-sensitivity (or lack thereof) may apply to the source database but not the target, or to the target but not the source.

● If the system or database is case-sensitive, Oracle GoldenGate supports the case sensitivity of database names, owner and schema names, object names, column names, and user names.

● If the system or database is case-insensitive (or is configured for case-insensitivity), Oracle GoldenGate converts all names to upper case.

To preserve case-sensitivity in an Oracle GoldenGate configuration

In Oracle GoldenGate parameter files, specify case-sensitive names exactly as they appear in the database. In TABLE and MAP parameters, enclose case-sensitive names in double quotes if the other database (the source or target of the case-sensitive objects) is not case-sensitive.

If replicating from a case-insensitive source to a case-sensitive target, enter the source names in upper case in the Replicat MAP statements, to reflect the fact that Extract writes them to the trail as uppercase.

For example:

```
MAP SALES.CUSTOMER, TARGET "Sales.Account";
```

Supported characters

Oracle GoldenGate supports alphanumeric characters in the names of objects, key columns, and non-key columns. Oracle GoldenGate also supports the following characters in columns that are not being used by Oracle GoldenGate as a key.

Table 1   Supported non-alphanumeric characters in object names and non-key column names[1]

Character	Description
~	Tilde
< >	Greater-than and less-than symbols
/	Forward slash
\	Backward slash
!	Exclamation point
@	At symbol
#	Pound symbol
$	Dollar symbol

Table 1   Supported non-alphanumeric characters in object names and non-key column names[1]

Character	Description
%	Percent symbol
^	Caret symbol
( )	Open and close parentheses
_	Underscore
-	Dash
+	Plus sign
=	Equal symbol
|	Pipe
[ ]	Begin and end brackets
{ }	Begin and end curly brackets (braces)

[1] The type of key that is being used by Oracle GoldenGate depends on the definition of a given table and whether there are any overrides by means of a KEYCOLS clause. Oracle GoldenGate will use a primary key, if available, or a unique key/index (selection is dependent on the database). In the absence of those definitions, all columns of the table are used, but a KEYCOLS clause overrides all existing key types. For columns that are being used by Oracle GoldenGate as a key, the characters in the names must be valid for inclusion in a WHERE clause. This list is all-inclusive; a given database platform may or may not support all listed characters.

Non-supported characters

Oracle GoldenGate does not support the following characters in object or column names.

Table 2   Non-supported characters in object and column names[1]

Character	Description
&	Ampersand
*	Asterisk
?	Question mark
:	Colon
;	Semi-colon
,	Comma
' '	Single quotes
" "	Double quotes

Table 2   Non-supported characters in object and column names[1]

Character	Description
'	Accent mark (Diacritical mark)
.	Period
	Space

[1] This list is all-inclusive; a given database platform may or may not support all listed characters.

CHAPTER 2
Installing Oracle GoldenGate

· · · · · · · · · · · · · ·

These instructions are for installing Oracle GoldenGate for the first time. To upgrade Oracle GoldenGate from one version to another, follow the instructions on:

http://www.oracle.com/technology/software/products/goldengate/index.html

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.

The installation process takes a short amount of time.

Installation overview

To install Oracle GoldenGate, the following steps are required:

Choosing an installation system for DB2

Choosing an installation system for DB2

Downloading Oracle GoldenGate

Setting library paths for dynamic builds on UNIX

Preparing to install Oracle GoldenGate within a cluster

Installing Oracle GoldenGate on Linux and UNIX

Installing Oracle GoldenGate on Windows

Integrating Oracle GoldenGate into a cluster

Choosing an installation system for DB2

To install Oracle GoldenGate for DB2 LUW, you can use either of the following configurations:

● Install Oracle GoldenGate on the DB2 database server. See one of the following:

"Installing Oracle GoldenGate on Linux and UNIX" on page 14

"Installing Oracle GoldenGate on Windows" on page 15

● Install Oracle GoldenGate on another server, and configure Oracle GoldenGate to connect remotely to the database server through DB2 Connect. All of the Oracle GoldenGate functionality that is supported for DB2 LUW is supported in this configuration. To use this option, proceed to "Choosing and configuring a system for remote capture or delivery".

Choosing and configuring a system for remote capture or delivery

In a remote installation, you install Oracle GoldenGate on a server that is separate from the source or target database server. This server can be any Linux, UNIX, or Windows platform that Oracle GoldenGate supports for the DB2 for LUW database. The Oracle GoldenGate build must match the version of DB2 LUW that is running on the remote installation server.

In this configuration, the location of the database is transparent to Extract and Replicat. Extract can read the DB2 logs on a source DB2 LUW database server, and Replicat can apply data to a target DB2 LUW server.

To configure remote capture or delivery

1. Install and run DB2 for LUW on the remote server that has DB2 Connect.

2. Catalog the remote server in the DB2 source or target database by using the following DB2 command:

    ```
    catalog tcpip node <db2_node_name> remote <remote_DNS_name>
    server <remote_port_number>
    ```

3. Add the DB2 source or target database to the DB2 catalog on the remote server by using the following DB2 command:

    ```
    catalog db <database_name> as <database_alias> at node <db_node_name>
    ```

    > **NOTE** Refer to the IBM DB2 LUW documentation for more information about these commands.

4. Download and install the Oracle GoldenGate build that is appropriate for DB2 database on the remote server.

Downloading Oracle GoldenGate

Download the appropriate Oracle GoldenGate build to each system that will be part of the Oracle GoldenGate configuration. To install Oracle GoldenGate on a remote server for connection through DB2 Connect, download the Oracle GoldenGate build that matches the version of DB2 LUW that is running on the remote server.

1. Navigate to http://edelivery.oracle.com.

2. On the **Welcome** page:
    ❍ Select your language.
    ❍ Click **Continue**.

3. On the **Export Validation** page:
    ❍ Enter your identification information.
    ❍ Accept the **Trial License Agreement** (even if you have a permanent license).
    ❍ Accept the **Export Restrictions**.
    ❍ Click **Continue**.

4. On the **Media Pack Search** page:

○ Select the **Oracle Fusion Middleware** Product Pack.

○ Select the platform on which you will be installing the software.

○ Click **Go**.

5. In the **Results List**:

○ Select the Media Pack that you want to download. *Note*: To install Oracle GoldenGate to connect to a DB2 target using DB2 Connect from a Windows or UNIX system, select the build that is labeled as **DB2Remote**; or to download Oracle GoldenGate to run on the DB2 target system, select the appropriate build for the database and platform. (See "Choosing an installation system for DB2" on page 10.)

○ Click **Continue**.

6. On the **Download** page:

○ Click **Download** for each component that you want. Follow the automatic download process to transfer the mediapack.zip file to your system.

> **NOTE**   Before installing the software, review the release notes for any new features, new requirements, or bug fixes that affect your current configuration. Review the readme file for known issues.

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 true ***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 libraries 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:

UNIX/Linux library path variables per platform

Platform[1]	Environment variable
IBM AIX	LIBPATH
IBM z/OS	
HP-UX	SHLIB_PATH
Sun Solaris	LD_LIBRARY_PATH[2]
HP Tru64 (OSF/1)	
LINUX	

[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.

Example

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

> **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.

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 is Oracle-certified.

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.
  - ❍ br
  - ❍ dirchk
  - ❍ dirdat
  - ❍ dirtmp

  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" on page 17 after you install Oracle GoldenGate.

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" on page 13 for more information.

Installing the Oracle GoldenGate files

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
   ```

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" on page 13 for more information.

Installing Oracle GoldenGate into a Windows Cluster

1. Log into one of the nodes in the cluster.

2. 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. 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.

Installing the Oracle GoldenGate files

1. Unzip the downloaded file(s) by using WinZip or an equivalent compression product.

2. 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.

3. From the Oracle GoldenGate folder, run the GGSCI program.

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

   ```
   CREATE SUBDIRS
   ```

5. Issue the following command to exit GGSCI.

   ```
   EXIT
   ```

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.

To specify a custom Manager name

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.

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.

To install Manager as a Windows service

1. (Recommended) Log on as the system administrator.

2. Click **Start** > **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 3   INSTALL options

Option	Description
ADDEVENTS	Adds Oracle GoldenGate events to the Windows Event Manager. By default, Oracle GoldenGate errors are generic. To produce more specific error content, copy the following files from the Oracle GoldenGate installation directory to the SYSTEM32 directory. category.dll ggsmsg.dll
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.

Table 3   INSTALL options (continued)

Option	Description
`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. (Windows Server 2008) 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.

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.

General requirements in 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 (such as Oracle Clusterware), 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

www.ntp.org or your systems administrator. See also the IOLATENCY option of the THREADOPTIONS parameter in the ***Oracle GoldenGate Windows and UNIX Reference Guide***.

4. Make certain that all nodes in the cluster have the same COMPATIBLE parameter setting.

5. When you configure Manager, add the AUTOSTART and AUTORESTART parameters so that Manager starts the replication processes automatically (see "Creating the Oracle GoldenGate instance" on page 46). You can, when needed, control Extract, Replicat, and other Oracle GoldenGate processes from within the Oracle GoldenGate user interfaces.

6. 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.

7. Configure Oracle GoldenGate as directed in this documentation.

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

CHAPTER 3
Preparing the system for Oracle GoldenGate

. . . .

Configuring the transaction logs for Oracle GoldenGate

To capture DML operations, Oracle GoldenGate reads the DB2 for LUW online logs by default, but will read the archived logs if an online log is not available. To ensure the continuity and integrity of Oracle GoldenGate processing, configure the logs as follows.

Retaining the transaction logs

Configure the database to retain the transaction logs for rollforward recovery by enabling one of the following parameter sets, depending on the database version.

DB2 LUW 9.1:

● Set the USEREXIT parameter to ON, which automatically sets LOGRETAIN to RECOVERY and forces a user exit program to archive and retrieve the log files.

● Set the LOGRETAIN parameter to RECOVERY, which retains the logs and enables them to be used for forward recovery.

DB2 LUW 9.5 and later:

Set the LOGARCHMETH parameters as follows:

● Set LOGARCHMETH1 to LOGRETAIN.

● Set LOGARCHMETH2 to OFF.

Alternatively, you can use any other LOGARCHMETH options, so long as forward recovery is enabled. For example, the following is valid:

● Set LOGARCHMETH1 to DISK.

● Set LOGARCHMETH2 to TSM.

To determine log retention parameters

1. Connect to the database.

   ```
   db2 connect to <database> user <username> using <password>
   ```

2. Get the database name.

   ```
   db2 list db directory
   ```

3. Get the database configuration for the database.

   ```
   db2 get db cfg for <database name>
   ```

The fields to view are:

```
Log retain for recovery status = RECOVERY
User exit for logging status = YES
```

To set log retention parameters

1. Issue one of the following commands.

To enable USEREXIT:

```
db2 update db cfg for <database name> using USEREXIT ON
```

If not using USEREXIT, use this command:

```
db2 update db cfg for <database name> using LOGRETAIN RECOVERY
```

To set LOGARCHMETH:

```
db2 update db cfg for <database name> using LOGARCHMETH1 LOGRETAIN
db2 update db cfg for <database name> using LOGARCHMETH2 OFF
```

2. Make a full backup of the database by issuing the following command.

```
db2 backup db <database> to <device>
```

3. Place the backup in a directory to which DB2 LUW has access rights. Contact your systems administrator if you get the following message:

```
SQL2061N An attempt to access media "device" is denied.
```

Specifying the archive path

Set the DB2 OVERFLOWLOGPATH parameter to the archive log directory. The node attaches automatically to the path variable that you specify.

To set the parameter

```
db2 connect to <database>
db2 update db cfg using overflowlogpath "<path>"
```

Exclude the node itself from the path. For example, if the full path to the archive log directory is /sdb2logarch/oltpods1/archive/OLTPODS1/NODE0000, the OVERFLOWLOGPATH value should be specified as /sdb2logarch/oltpods1/archive/OLTPODS1.

Preparing tables for processing

The following table attributes must be addressed in an Oracle GoldenGate environment.

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.

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.

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.

Using KEYCOLS to specify a custom key

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. For more information, see the Oracle GoldenGate *Windows and UNIX Reference Guide*.

Preventing key changes

Do not add columns to a key after Oracle GoldenGate starts extracting data from the table. This rule applies to a primary key, a unique key, a KEYCOLS key, or an all-column key. DB2 LUW does not supply a before image for columns that are added to a table. If any columns

in a key are updated on the source, Oracle GoldenGate needs a before image to compare with the current values in the target table when it replicates the update.

Enabling change capture

Configure DB2 to log data changes in the expanded format that is supplied by the DATA CAPTURE CHANGES feature of the CREATE TABLE and ALTER TABLE commands. This format provides Oracle GoldenGate with the entire before and after images of rows that are changed by update statements. You can use GGSCI to issue the ALTER TABLE command as follows.

To enable change capture from GGSCI

1. From the Oracle GoldenGate directory, run GGSCI.

2. Log on to DB2 from GGSCI as a user that has ALTER TABLE privileges. Specify the data source name with SOURCEDB and specify the user login with USERID and PASSWORD.

   ```
   DBLOGIN SOURCEDB <dsn>, USERID <user>[, PASSWORD <password>]
   ```

3. Issue the following command. where <table> is the fully qualified name of the table. You can use a wildcard to specify multiple table names but not owner names.

   ```
   ADD TRANDATA <table>
   ```

ADD TRANDATA issues the following command, which includes logging the before image of LONGVAR columns:

```
ALTER TABLE < name> DATA CAPTURE CHANGES INCLUDE LONGVAR COLUMNS;
```

To exclude LONGVAR logging

To omit the INCLUDE LONGVAR COLUMNS clause from the ALTER TABLE command, use ADD TRANDATA with the EXCLUDELONG option.

```
ADD TRANDATA <table>, EXCLUDELONG
```

> **NOTE**   If LONGVAR columns are excluded from logging, the Oracle GoldenGate features that require before images, such as the GETUPDATEBEFORES, NOCOMPRESSUPDATES, and NOCOMPRESSDELETES parameters, might return errors if tables contain those columns. For a workaround, see the REQUIRELONGDATACAPTURECHANGES | NOREQUIRELONGDATACAPTURECHANGES options of the TRANLOGOPTIONS parameter.

Maintaining materialized query tables

To maintain parity between source and target materialized query tables (MQT), you replicate the base tables, but not the MQTs. The target database maintains the MQTs based on the changes that Replicat applies to the base tables.

The following are the rules for configuring these tables:

- Include the base tables in your TABLE and MAP statements.
- Do not include MQTs in the TABLE and MAP statements.
- Wildcards can be used in TABLE and MAP statements, even though they might resolve MQT names along with regular table names. Oracle GoldenGate automatically excludes MQTs from wildcarded table lists. However, any MQT that is explicitly listed in an Extract TABLE statement by name will cause Extract to abend.

Preparing for initial extraction

During the initialization of the Oracle GoldenGate environment, you will be doing an initial data synchronization and starting the Oracle GoldenGate processes for the first time. In conjunction with those procedures, you will be creating process groups. To create an Extract group, an initial start position must be established in the transaction log. This initial read position will be on a transaction boundary that is based on one of the following:

- a timestamp
- the end of the transaction file
- a specific LSN value

The start point is specified with the BEGIN option of the ADD EXTRACT command.

When Extract starts for the first time, it captures all of the transaction data that it encounters after the specified start point, but none of the data that occurred *before* that point. This can cause partial transactions to be captured if open transactions span the start point.

To ensure initial transactional consistency

To avoid the capture of partial transactions, initialize the Extract process at a point in time when the database is in a quiesced state. DB2 provides a QUIESCE command for such a purpose. This is the only way to ensure transactional consistency.

> **NOTE** After Extract is past the initialization, subsequent restarts of Extract will not extract partial transactions, because the process uses recovery checkpoints to mark its last read position.

To view open transactions

IBM provides a utility called db2pd for monitoring DB2 databases and instances. You can use it to view information about open transactions and to determine if any of them span the start point. However, because DB2 LUW log records lack timestamps, it might not be possible to make an accurate assessment. If possible, quiesce the database prior to initialization of Oracle GoldenGate.

Specifying the DB2 LUW database in parameter files

For an Oracle GoldenGate process to connect to the correct DB2 LUW database, you must specify the database with the following parameters:

- ❍ Specify the DB2 source database with the Extract parameter SOURCEDB.
- ❍ Specify the DB2 target database name with the Replicat parameter TARGETDB.

These parameters must be used whether you are running Oracle GoldenGate locally on the source or target DB2 LUW server, or on a remote server where DB2 Connect is installed. For more information about these parameters, see the Oracle GoldenGate *Windows and UNIX Reference Guide*.

Character set considerations

Oracle GoldenGate provides globalization support that is documented in the Oracle GoldenGate *Windows and UNIX Administrator's Guide*. As part of this support, Oracle GoldenGate supports character-set conversion. Character set conversion is required when the source character set differs from the target character set. Replicat performs this conversion automatically in most cases. However there are some cases in which a specific parameter specification is required for Extract on a DB2 z/OS source to facilitate the appropriate conversion by Replicat.

DB2 for z/OS installed as MIXED=NO

(All character strings are single-byte data)

In this case, multi-byte data will not exist in either ASCII or EBCDIC source tables. If the target environment is multi-byte, the automatic character conversion by Replicat does not support a mix of ASCII and EBCDIC source data. Therefore, Extract must either convert all ASCII data to EBCDIC or all EBCDIC data to ASCII. This conversion assumes U.S. English ASCII and EBCDIC character sets. The parameters that control this are TRAILCHARSETASCII and TRAILCHARSETEBCDIC.

- In a mixed=no source environment, TRAILCHARSETASCII causes Extract to convert data from any EBCDIC table to ASCII before it is written to the trail file.
- In a mixed=no source environment, TRAILCHARSETEBCDIC causes Extract to convert data from any ASCII table to EBCDIC before it is written to the trail file.

DB2 for z/OS installed as MIXED=YES

(Character strings can be either single-byte data or mixed single-byte and multi-byte data)

The automatic character conversion by Replicat does not support a mix of multi-byte ASCII data and multi-byte EBCDIC data. Therefore, any given Extract cannot process both ASCII and EBCDIC tables from a mixed=yes source environment. Two separate Extracts must be used to process both ASCII and EBCDIC tables from a mixed=yes source, if that is required. The parameters that control this are TRAILCHARSETASCII and TRAILCHARSETEBCDIC.

- In a mixed=yes source environment, TRAILCHARSETASCII causes Extract to ensure that only ASCII tables are processed. Extract fails with an error if data from an EBCDIC table is encountered.
- In a mixed=yes source environment, TRAILCHARSETEBCDIC causes Extract to ensure that only EBCDIC tables are processed. Extract fails with an error if data from an ASCII table is encountered.

In all cases, data from Unicode tables can be processed correctly without any special consideration and regardless of the specification for TRAILCHARSETASCII or TRAILCHARSETEBCDIC.

CHAPTER 4
Uninstalling Oracle GoldenGate

· · · · · · · · · · · · · ·

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.

Uninstalling Oracle GoldenGate from Linux or UNIX

On all systems:

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.

On a source system:

7. Log into the database with the DBLOGIN command, and then run the DELETE TRANDATA command to alter the tables to DATA CAPTURE NONE.

On any system where a Replicat checkpoint table is being used:

8. Log into the database with the DBLOGIN command, and then remove the Replicat checkpoint table by running the DELETE CHECKPOINTTABLE command.

On all systems:

9. Make certain all processes are stopped (including GGSCI).

10. Remove the Oracle GoldenGate files by removing the installation directory.

Removing Oracle GoldenGate from Windows Cluster

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. Delete the CATEGORY.DLL and GGSMSG.DLL files from the Windows SYSTEM32 folder.

8. Move the cluster group to the next node in the cluster, and repeat from step 4.

9. Follow the instructions in "Uninstalling Oracle GoldenGate from Windows (non-
    cluster)".

Uninstalling Oracle GoldenGate from Windows (non-cluster)

On all systems:

1. (Suggested) 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** > **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. Delete the CATEGORY.DLL and GGSMSG.DLL files from the Windows SYSTEM32 folder.

On a source system:

9. Log into the database with the DBLOGIN command, and then run the DELETE TRANDATA
    command to alter the tables to DATA CAPTURE NONE.

On any system where a Replicat checkpoint table is being used:

10. Log into the database with the DBLOGIN command, and then remove the Replicat
    checkpoint table by running the DELETE CHECKPOINTTABLE command.

On all systems:

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

APPENDIX 1
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.

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 4   Programs and utilities

Program	Description
cobgen	Generates source definitions based on COBOL layouts. Used for Oracle GoldenGate for Datawise on Stratus.
convchk	Converts checkpoint files to a newer version.
ddlcob	Generates target DDL table creation statements based on COBOL layouts. Used for Oracle GoldenGate for Datawise on Stratus.
ddlgen	Generates target database table definitions based on source database DDL. Used primarily on the NonStop platform.
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.
ggsci	User interface to Oracle GoldenGate for issuing commands and managing parameter files.

· ·

Table 4   Programs and utilities (continued)

Program	Description
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.
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.

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 5   Subdirectories

Directory	Description
br	Contains the checkpoint files for the bounded recover feature.
cfg	Contains the property and xml files that are used to configure Oracle GoldenGate Monitor.
dirdb	Contains the datastore 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.

Table 5   Subdirectories (continued)

Directory	Description
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
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
dirjar	Contains the Java executable files that support Oracle GoldenGate Monitor.
dirout	This directory is not used any more.

Table 5   Subdirectories (continued)

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 the Oracle Wallet that supports Oracle GoldenGate Monitor. This directory is not installed until the utility that creates the wallet is run.
UserExitExamples	Contains sample files to help with the creation of user exits.

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 6   Other 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 program.
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_access.tpl	Template used by the DDLGEN utility to convert source DDL to Microsoft Access DDL.
ddl_cleartrace.sql	Script that removes the DDL trace file. (Oracle installations)
ddl_db2.tpl	Template used by the DDLGEN utility to convert source DDL to DB2 DDL (Linux, UNIX, Windows).
ddl_db2_os390.tpl	Template used by the DDLGEN utility to convert source DDL to DB2 DDL (z/OS systems).
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_informix.tpl	Template used by the DDLGEN utility to convert source DDL to Informix DDL.

Table 6   Other files (continued)

Component	Description
ddl_mss.tpl	Template used by the DDLGEN utility to convert source DDL to SQL Server DDL.
ddl_mysql.tpl	Template used by the DDLGEN utility to convert source DDL to MySQL DDL.
ddl_nopurgeRecyclebin.sql	Empty script file for use by Oracle GoldenGate support staff.
ddl_nssql.tpl	Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.
ddl_ora9.sql ddl_ora10.sql ddl_ora11.sql ddl_ora10upCommon.sql	Scripts that run programmatically as part of Oracle GoldenGate DDL support; do not run these scripts.
ddl_oracle.tpl	Template used by the DDLGEN utility to convert source DDL to Oracle DDL.
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_sqlmx.tpl	Template used by the DDLGEN utility to convert Tandem Enscribe DDL to NonStop SQL/MX DDL.
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)
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_sybase.tpl	Template used by the DDLGEN utility to convert source DDL to Sybase DDL.

Table 6   Other files (continued)

Component	Description
ddl_tandem.tpl	Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.
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>_create.sql demo_more_<db>_create.sql demo_<db>_insert.sql demo_more_<db>_insert.sql demo_<db>_lob_create.sql demo_<db>_misc.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.

Table 6   Other files (continued)

Component	Description
icudt38.dll icuin38.dll icuuc38.dll	Windows shared libraries for International Components for Unicode.
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.
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.
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)

Table 6   Other files (continued)

Component	Description
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.

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 the type of database.

Do not change the names or attributes of the columns in this table. You can change table storage attributes as needed.

Table 7   Checkpoint table definitions

Column	Description
GROUP_NAME (primary key)	The name of a Replicat group using this table for checkpoints. There can be multiple Replicat groups using the same table.
GROUP_KEY (primary key)	A unique identifier that, together with GROUPNAME, uniquely identifies a checkpoint regardless of how many Replicat groups are writing to the same table.
SEQNO	The sequence number of the checkpoint file.
RBA	The relative byte address of the checkpoint in the file.

Table 7   Checkpoint table definitions

Column	Description
AUDIT_TS	The timestamp of the checkpoint position in the checkpoint file.
CREATE_TS	The date and time when the checkpoint table was created.
LAST_UPDATE_TS	The date and time when the checkpoint table was last updated.
CURRENT_DIR	The current Oracle GoldenGate home directory or folder.

Index

• • • • • • • • • • • • • •

• •