Oracle® Fusion Middleware

Installing and Configuring Oracle GoldenGate for Informix

12*c* (12.2.0.1)

E66366-04

February 2017

This document describes how to install and set up for Oracle GoldenGate for Informix.

ORACLE®

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

E66366-04

Contents

3 Preparing the System for Oracle GoldenGate

4 Configuring Oracle GoldenGate for Capture

5 Configuring Oracle GoldenGate for Apply

6 Adding Process Groups

7 Uninstalling Oracle GoldenGate

A   Oracle GoldenGate Installed Components

Preface

With Oracle GoldenGate for Informix, you can replicate data to and from supported Informix Dynamic Server databases or between an Informix Dynamic Server database and a database of another type. Oracle GoldenGate for Informix supports data filtering, mapping, and transformation unless noted otherwise in this documentation.

This guide helps you get started with installing Oracle GoldenGate on an Informix Dynamic Server 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.

This chapter includes the following sections:

- Section 1.1, "Verifying Certification and System Requirements"
- Section 1.2, "Operating System Requirements"
- Section 1.3, "Informix Dynamic Server Requirements"
- Section 1.4, "Supported Informix Dynamic Server Data Types"
- Section 1.5, "Non-Supported Informix Dynamic Server Data Types"
- Section 1.6, "Supported Objects and Operations for Informix Dynamic Server"
- Section 1.7, "Non-Supported Objects and Operations for Informix Dynamic Server"

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 Informix Dynamic Server operating system requirements. These requirements fall into one of the following categories:

- Section 1.2.1, "Memory Requirements"
- Section 1.2.2, "Disk Requirements"
- Section 1.2.3, "Temporary Disk Requirements"
- Section 1.2.4, "Network"
- Section 1.2.5, "Operating System Privileges"
- Section 1.2.6, "Other Programs"

1.2.1 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 *Administering Oracle GoldenGate for Windows and UNIX*

1.2.2 Disk Requirements

Assign free disk space according to the following instructions:

- 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 *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 the *Administering Oracle GoldenGate for Windows and UNIX* 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.

The Oracle GoldenGate Manager process requires the following operating system privileges:

- Full control over the files and folders within the Oracle GoldenGate directories.

- Full control over the trail files, if stored in a location other than the Oracle GoldenGate directory.

- Membership in the local Administrators Group.

The programs that capture and replicate data (Extract and Replicat) run under the Manager account and inherit those Administrator rights.

The Manager process can run as a Windows service, or it can run interactively as the current user. You make this decision when installing Oracle GoldenGate. See Section 2.4, "Installing Oracle GoldenGate on Windows" for more information.

1.2.6 Other Programs

- 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.

- Make sure an Open Database Connectivity (ODBC) driver manager is installed on the system. On UNIX, you can use the `unixODBC` driver manager, which is free for download at

  http://www.unixodbc.org/

1.3 Informix Dynamic Server Requirements

To operate with Informix Dynamic Server (IDS) database, Oracle GoldenGate requires the following setup in the database instance.

Oracle recommends that you install the latest Informix Client SDK (CSDK) rather than using the CSDK that is delivered with the IDS.

http://www-01.ibm.com/software/data/informix/

1.3.1 Install the Informix Client SDK Fix Pack

You must install the Informix CSDK version 4.10 fix to any of the supported Informix versions you have installed prior to installing Oracle GoldenGate. This is because Oracle GoldenGate was developed with this fix and failure to install it results in unpredictable Extract and Replicat behavior and may abend. The Informix Server fixes are found at:

http://www-933.ibm.com/support/fixcentral/

In addition, it was determined that there are memory leak issues in the IDS that IBM has addressed. Oracle recommends that you upgrade your databases to the versions as appropriate for your environment:

Existing IDS Version	Upgrade to Version
12.10	12.10.FC3W1
11.70	11.70.xC8
11.50	No upgraded needed.

1.3.2 Database Configuration

The database should be configured according to the following recommendations:

- A root installation of Informix Dynamic Server must exist on a system that is a source or target for Oracle GoldenGate. You can install Oracle GoldenGate on any supported platform that is remote from the system hosting the Informix database.

- The database libraries `libthcli`, `libifdmr`, `libifgls.so`, and `libifglx.so` runtime libraries must exist on the system where Oracle GoldenGate is installed (source, target, or remote system). The `LD_LIBRARY_PATH` or `LIBPATH` variable (depending on the platform) must be set to the installation path of these libraries. As an example, assuming `$INFORMIXDIR` is the path to the root Informix installation, the library path for the runtime environment must be set as follows:

  ```
  LD_LIBRARY_PATH=$INFORMIXDIR/lib/cli:$INFORMIXDIR/lib/esql:$ LD_LIBRARY_PATH
  ```

- The Informix Dynamic Server database must be running when installing and running Oracle GoldenGate on any Informix system.

- A `SYSCDC` database must be created before you install Oracle GoldenGate on a source or target Informix system. The script to create the `SYSCDC` database and related functions typically exists in the location `$INFORMIXDIR/etc/syscdcv1.sql`. Run the script as the `informix` user.

- Oracle GoldenGate passes change data to the Extract process on an Informix source system. This is enabled for Informix database objects by means of the Oracle GoldenGate `ADD TRANDATA` command.

- Enable transaction logging by any of the three modes supported by Informix: Unbuffered, buffered, or ANSI-Compliant. This must be done for both the source and target Informix databases. You can use the Informix tool `ondblog` that is included in the Informix Dynamic Server installation to change the logging.

- Set an appropriate value for the IDS logical log size by using the `onmode` tool.

```
onmode -wf logsize=value
```

To determine the optimal log size, you can get performance data from the database engine by using the following command:

```
onstat -g ckp
```

■ Support for the Informix Web DataBlade HTML data type is reliant on having the IBM Informix Web DataBlade module installed and enable. Further, the source or target database must have the Web DataBlade module registered on both databases as in the following example:

```
EXECUTE FUNCTION
sysbldprepare('web.4.13.FC4','create');
```

■ Set the GL_USEGLU environment variables as follows:

```
SET GL_USEGLU=0
```

■ Optionally, install and configure Open Admin (OAT) to connect to the Informix server instance.

1.3.3 Database Connection

Oracle GoldenGate uses the Informix ODBC driver to connect to the SYSCDC database. The driver files are libthcli or libifdmr on UNIX and iclit09b.so on Windows. The driver version must be compatible with the Informix Dynamic Server version.

The Oracle GoldenGate Extract process must connect as the informix user, and only to the SYSCDC database. This user has all of the database privileges that are required by Oracle GoldenGate to capture data from an Informix database. The credentials of this user must be supplied in the Oracle GoldenGate parameter files in a secure manner. You will be prompted for this information during the configuration instructions.

1.4 Supported Informix Dynamic Server Data Types

Oracle GoldenGate supports most Informix data types except those listed under Section 1.5, "Non-Supported Informix Dynamic Server Data Types."

Limitations of Support

■ Oracle GoldenGate does not support the filtering, column mapping, or manipulation of large objects larger than 4K. Full Oracle GoldenGate functionality can be used for objects that are 4K or smaller.

■ LOB objects are not captured through the API. To capture a LOB column, the FETCHCOLS option must be used in the Extract TABLE statement that specifies the affected table. For more information about this parameter, see *Reference for Oracle GoldenGate for Windows and UNIX*.

■ Oracle GoldenGate may truncate an INTERVAL data type to no more than four digits if that INTERVAL has a non-default precision.

■ Oracle GoldenGate supports Informix DATETIME 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 supports Informix replication onto HTML data type column only if a function named, textashtml, is created on the target database as in the following example:

```
create function textashtml(lo_text REFERENCES TEXT) returning html;
return lo_text::clob::html;
end function
```

  You must ensure that you grant Execute privileges to users that will invoke this function. When replicating HTML data type columns, you must also ensure that the data size does not exceed 65534 bytes if there is more than one HTML column in the same table due to a limitation in the Informix ODBC driver.

- To replicat LOBs, you need to create the following procedures on the target:

```
create function textasclob(lo_text REFERENCES TEXT) returning clob;
return lo_text::clob;
end function;

create function textasclob(lo_clob CLOB) returning clob;
return lo_clob;
end function;

create function byteasblob(lo_byte REFERENCES BYTE) returning blob;
return lo_byte::blob;
end function;

create function byteasblob(lo_blob BLOB) returning blob;
return lo_blob;
end function;
```

  You must ensure that you grant Execute privileges to users that will invoke these function.

- Oracle GoldenGate supports data containing 0x00 only when you use he BYTE data type because Informix interprets zero values as terminators.

- The MONEY type can be declared as COLMONEY MONEY, without scale and precision. However, without specifying the scale and precision, Replicat cannot return the same value with the same precision and scale across every locale. For example, in the Japanese locale, when you enter 3.1 it returns \3 because the backslash symbol is the ASCII representation for the Yen symbol and the number is rounded because there are no decimal representations for Yen. This is expected behavior because you cannot prevent Informix from rounding the number in Japanese locale.

  Oracle recommends that you define the MONEY column explicitly with precision and scale as in the following example:

```
COLMONEY MONEY(p,s)
```

  In doing so, you can replicate all the digits after the decimal point irrespective of the locale.

- For the DECIMAL and NUMERIC data types, Informix only supports a maximum default precision to 16 digit so you must ensure the number of digits do not exceed this maximum if specified during the table creation. When this default precision value is exceeded, data loss is possible.

1.5 Non-Supported Informix Dynamic Server Data Types

The following Informix data types are not supported are not supported by Oracle GoldenGate.

- All user-defined types, except HTML, are unsupported.

- Collection data types (SET, MULTISET, LIST, ROW)

- Spatial data type not supported for Informix.

1.6 Supported Objects and Operations for Informix Dynamic Server

The following objects and operations are supported:

- Oracle GoldenGate supports the capture and apply of DML operations on tables that contain rows of up to the maximum length supported by the database engine.

- The following operations are supported:

  - INSERT

  - UPDATE

  - DELETE

  - TRUNCATE

    > **Note:** While TRUNCATE is supported, Oracle does not recommend the use of this operation because it may cause performance issues.

  - ROLLBACK of a transaction

  - ROLLBACK to a savepoint in a transaction

- Oracle GoldenGate supports Informix tables that use data compression and column encryption.

- Oracle GoldenGate Extract (capture) processes support capture from multiple Informix databases.

- Oracle GoldenGate Extract processes support capture from fragmented tables. The default behavior of the UPDATE operation on a fragmented source table that results in modification of a key column (primary key) is that the capture converts the update into the following operations in order:

```
INSERT new row
DELETE old row
```

This behavior is in accordance with the way the Informix logical logs are written for primary key update on fragmented tables.

- Oracle GoldenGate Replicat processes always perform a DELETE operation on the target database when the source operation (in the trail) is TRUNCATE.

- Oracle GoldenGate only supports tables and columns that have lower-case names. If mixed or upper-case table names exist in your database, you must create synonyms in lower case for these objects. Column names that are not lowercase, are not supported. Extract will not capture from tables or columns that have mixed or upper-case names.

  For example, if you want Extract to capture from a table named UPPERCASETAB, issue the following command to create a lower-case synonym:

```
CREATE SYNONYM 'mytables'.tabalias1 for 'mytables'."UPPERCTAB"
```

In the Oracle GoldenGate parameter files, specify only the synonym when listing and mapping tables. Only the synonym will be referenced by the Oracle GoldenGate processes and stored in the trails and other files.

- Oracle GoldenGate does not support wildcarding at the beginning of a database catalog part. For example, `*.schema_name_table_name` is not supported.

- The filtering of Replicat transactions in an active-active configuration is supported based on the user ID that is assigned to the specific Replicat process by the operating system. To determine what a user's name is, do the following:

  - On UNIX, use the following command:

    ```
    id informix
    uid=501(informix) gid=501(informix) groups=501(informix),0(root)
    context=user_u:system_r:unconfined_t
    ```

  - On Windows, you will need to query the following registry key to retrieve the `DOMAIN\username or MACHINE\username associated with "UID` user ID:

    ```
    HKEY_LOCAL_
    MACHINE\SOFTWARE\Informix\OnLine\%INFORMIXSERVER%\Security\Users\Usernames\
    UID#
    ```

    Note: `"%INFORMIXSERVER%"` in the preceding syntax is the value of the `INFORMIXSERVER` environment variable for that instance. `"UID#"` is the literal text "UID" with the numeric user ID concatenated onto the end. For example, in the value `UID23`, the 23 is the user ID from the log record.

- When Replicat is used to replicate data to multiple target databases in the same process, the `DBOPTIONS SPTHREAD` parameter and option must be used in the Replicat parameter file. For other uses of `SQLEXEC`, this parameter is not mandatory.

- Oracle GoldenGate does not support the use of the `SQL_LONGVARCHAR` (a native data type for `LVARCHAR`) parameter as a key column in either Extract or Replicat operations. For example, if you create `SQL_LONGVARCHAR` as key column in your replication table, then Oracle GoldenGate will not treat it as key column and will consider other pseudo columns as part of the key columns.

- Reading of transaction records from the archive log is not supported.

- Dynamic wildcarding resolution is not supported.

- Limitations on Automatic Heartbeat Table support are as follows:

  - The `GLOBALS` file must have a three-part table name, which is the database name, the schema name, and the heartbeat table name all separated by a period (.). For example, `srcdb.informix.gg_heartbeat`. The schema name must be a operating system user and an Informix DBA/Administrator. Typically, "informix" is used for the schema name.

  - A database login to the 'syscdcv1' database is mandatory before issuing any heartbeat table commands.

  - The default Oracle GoldenGate schema, `GGSUSER`, is not supported. Informix CDC works with a user that is an operating system user and an Informix DBA/Administrator, which is typically called "informix". Informix CDC

requires a three-part table name including a catalog name, a schema name, and a table name.

- For non-Informix user/schema, all the connect and DBA privileges to sysadmin and sysmaster system databases must be applied. For example, an Oracle GoldenGate installation with a ggatsuser1 user/schema, the GLOBALS entry for heartbeat would be:

```
heartbeat_table srcdb.ggatsuser1.gg_heartbeat
```

If the non-Informix user/schema does *not* have the required privileges, the heartbeat feature cannot operate correctly.

- Mazovia character set is supported in Informix as CP437. You must ensure that you observe following when you use Mazovia locale in your source database.

    - If the Mazovia locale is setup in the source Informix and the target has UTF-8 or other on any heterogeneous database, then the CHARMAP override feature should be used to add 17 code points of Mazovia charset that does not exist in the CP437. Then these code points should be mapped to corresponding target character set code point to avoid any data loss in the target.

    - If the source is the Informix Mazovia character set and target has oracle UTF-8 character set then both CHAR and NCHAR in the same scenario should not be used. Ensure that you do not add a mix of CHAR and NCHAR code points in your CHARMAP map because this will not work and it may corrupt the data. For example, the following is not supported:

```
COLMAP(CHAR1= informix_char, NCHAR1=informix_nchar)
```

For the same scenario you cannot have 2 CHARMAP files having UTF-8 and UTF-16 code points together; you can only have one or the other code point in the CHARMAP file. Mixed of UTF-8 and UTF-16 is not supported.

Informix N(VAR)CHAR/(VAR)CHAR/LVARCHAR/TEXT to Oracle VARCHAR2/CHAR/CLOB is supported.

Oracle NVARCHAR2/NCHAR/NCLOB is *not* supported.

    - If the source has the Informix Mazovia charset and the target has also Informix Mazovia character set, then CHARMAP is not needed. For example:

```
N(VAR)CHAR/(VAR)CHAR/LVARCHAR/TEXT or CHAR/N(VAR)CHAR
```

1.7 Non-Supported Objects and Operations for Informix Dynamic Server

The following objects and operations are not supported by Informix Dynamic Server:

- All system databases are excluded from capture by Oracle GoldenGate and will be ignored if they are specified in the parameter files. These are: sysadmin, syscdcv1, sysmaster, sysuser, and sysutils.

- All tables that have a table identifier (tabid) of less than, or equal to, 99 are excluded from capture by Oracle GoldenGate. These are Informix system tables.

- Capture and apply of data definition language (DDL) operations is not supported.

- Capture from log archives or backups is not supported. The transaction logs must be online. See Chapter 3, "Preparing the System for Oracle GoldenGate" for instructions on sizing the logs so that they remain available until Extract is finished processing them.

- Extract can be started at any valid log sequence number (LSN) that is present in the active logical logs. Extract cannot be positioned at an LSN that exists in an archive or backup log. For more information about positioning options, see *Reference for Oracle GoldenGate for Windows and UNIX*. For more information about Extract repositioning options, see *Reference for Oracle GoldenGate for Windows and UNIX*.

- The `WILDCARDRESOLVE` parameter must be set to `IMMEDIATE` in the parameter files.

- The `ATCSN` and `AFTERCSN` options are not supported in any IDS version.

- Do not attempt to process two Replicat operations using the same trail to Replicate sharing the same HB table on the same database as this results in an error.

2

Installing Oracle GoldenGate

This chapter contains the procedures installing Oracle GoldenGate for Informix Dynamic Server. It contains the following sections:

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 *Upgrading Oracle GoldenGate for Windows and UNIX*.

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.

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 Installing Oracle GoldenGate on Linux and UNIX

Follow these steps to install Oracle GoldenGate for Oracle on a Linux or UNIX system. You can install Oracle GoldenGate directly on the server where the Informix Dynamic Server is installed, or you can install it on any server that has an installation of the ODBC driver libraries. For more information, see Section 1.3.3, "Database Connection."

This procedure involves the following steps:

Section 2.3.1, "Setting Oracle GoldenGate Library Variables"

Section 2.3.2, "Installing the Oracle GoldenGate Files"

2.3.1 Setting Oracle GoldenGate Library Variables

Oracle GoldenGate uses shared libraries. When you install Oracle GoldenGate on a UNIX or Linux 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_
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 variables shown in Table 2–1:

Table 2–1   UNIX/Linux Library Path Variables Per Platform

Platform	Environment variable
Sun Solaris	LD_LIBRARY_PATH
LINUX	

Example

```
export LD_LIBRARY_PATH=/ggs/11.0:$LD_LIBRARY_PATH
```

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

2.3.2 Installing the Oracle GoldenGate Files

To install the Oracle GoldenGate files, do the following:

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
   ```

2.4 Installing Oracle GoldenGate on Windows

Follow these steps to install Oracle GoldenGate for Oracle on a Windows system. This procedure involves the following steps:

Section 2.4.1, "Installing the Oracle GoldenGate Files"

Section 2.4.2, "Specifying a Custom Manager Name"

Section 2.4.3, "Installing Manager as a Windows Service"

2.4.1 Installing the Oracle GoldenGate Files

To install the Oracle GoldenGate files, do the following:

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`

6. Copy the following files from the Oracle GoldenGate installation directory to the `SYSTEM32` directory.

   ```
   category.dll
   ggsmsg.dll
   ```

2.4.2 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.

To specify a custom Manager name, use this procedure:

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.

2.4.3 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.

To install Manager as a Windows service

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

2. Click **Start**, then **Run**, and then 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` utility with the following syntax:

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

Where `option` is one of the following:

Table 2–2 `INSTALL` Utility 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 the `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` utility to run.

The `INSTALL` utility 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 returns a prompt.

3

Preparing the System for Oracle GoldenGate

This chapter contains steps to take so that the database with which Oracle GoldenGate interacts is configured properly to support Oracle GoldenGate capture and replication. Some steps apply to just a source system, some just to a target, and some to both.

This chapter is comprised of the following sections:

- Section 3.1, "Configuring an ODBC Database Connection"
- Section 3.2, "Preparing Tables for Processing"
- Section 3.3, "Preparing the Logical Logs for Capture"
- Section 3.4, "Supporting the Capture of LOBs"

3.1 Configuring an ODBC Database Connection

Oracle GoldenGate connects to an Informix Dynamic Server database through Open Database Connectivity (ODBC). Follow these instructions to create an Informix Dynamic Server system data source name (DSN). A DSN stores information about how to connect to an Informix Dynamic Server database through ODBC. Create a DSN for each Informix Dynamic Server source and target database.

3.1.1 Configuring Language Settings

The Oracle GoldenGate apply process (Replicat) supports the conversion of data from one character set to another when the data is contained in character column types. To enable this support, make certain to properly set the language parameters when configuring ODBC, as follows:

```
CLIENT_LOCALE=en_us.8859-1
```

```
DB_LOCALE=en_us.8859-1
```

For additional support information and limitations, see *Administering Oracle GoldenGate for Windows and UNIX*.

3.1.2 Configuring ODBC on UNIX

These instructions configure and test ODBC on a UNIX system.

3.1.2.1 Downloading an ODBC Driver Manager

The Informix ODBC driver on UNIX requires an ODBC Driver Manager. A common ODBC Driver Manager is available for free from the UNIX ODBC Project web site at

http://www.unixodbc.org/

It needs to be compiled from the source. Make certain that you download a version that is 2.3 or later.

If you want to compile the ODBC Driver Manager from the source files, make certain to compile it as 64-bit if the Informix ODBC Driver is a 64-bit version. This requires you to set the environment variable ' CFLAGS="-DBUILD_REAL_64_BIT_MODE" ' before running `configure`, `make`, and `make install`.

3.1.2.2 Configuring the `odbc.ini` File

On UNIX, you need to configure an `odbc.ini` file.You can use the one shipped with the Informix installation. It is `$INFORMIXDIR/etc/odbc.ini`. Configure a DSN according to the following example.

```
;------------------------------------------------------------------------
; IBM INFORMIX ODBC Sample File
;
; File:odbc.ini
;
;------------------------------------------------------------------------
[ODBC Data Sources]
Infdrv1=IBM INFORMIX ODBC DRIVER
Infdrv2=IBM INFORMIX ODBC DRIVER
;
; Define ODBC Database Driver's Below - Driver Configuration Section
;
[Infdrv1]
Driver=/home/informix/ifx1170/lib/cli/iclit09b.so
Description=IBM INFORMIX ODBC DRIVER
Database=stores_demo
LogonID=odbc
pwd=odbc
Servername=ids_server1
[Infdrv2]
Driver=/home/informix/ifx1170/lib/cli/iclis09b.so
Description=IBM INFORMIX ODBC DRIVER
Database=stores_demo
LogonID=odbc
pwd=odbc
Servername=ids_server2
CursorBehavior=0
CLIENT_LOCALE=en_us.8859-1
DB_LOCALE=en_us.8859-1
TRANSLATIONDLL=/home/informix/ifx1170/lib/esql/igo4a304.so
;
; UNICODE connection Section
;
; adding here new ODBC DSN
[superstores]
Driver=/home/informix/ifx1170/lib/cli/iclis09b.so
Database=superstores_demo
Servername=ol_informix1170
CursorBehavior=0
CLIENT_LOCALE=en_us.8859-1
DB_LOCALE=en_us.8859-1
TRANSLATIONDLL=/home/informix/ifx1170/lib/esql/igo4a304.so
[ODBC]
;uncomment the below line for UNICODE connection
;UNICODE=UCS-4
;
; Trace file Section
```

```
;
Trace=0
TraceFile=/tmp/odbctrace.out
InstallDir=/home/informix

TRACEDLL=idmrs09a.so
```

3.1.2.3 Configuring the `odbcinst.ini` File

You also need to add content to the `$INFORMIXDIR/etc/odbcinst.ini` file, as shown in the following example.

```
;-------------------------------------------------------------------------
; IBM INFORMIX ODBC Sample File
;
; File: odbcinst.ini
;
;-------------------------------------------------------------------------
[ODBC Drivers]
IBM INFORMIX ODBC DRIVER=Installed
[IBM INFORMIX ODBC DRIVER]
Driver=/home/informix/ifx1170/lib/cli/iclit09b.so
Setup=/home/informix/ifx1170/lib/cli/iclit09b.so
APILevel=1
ConnectFunctions=YYY
DriverODBCVer=03.51
FileUsage=0
SQLLevel=1
smProcessPerConnect=Y
```

3.1.2.4 Testing the ODBC Configuration

To test the ODBC configuration, make certain that the `INFORMIX` environment variable is set and that the ODBC libraries are set, and then issue the following commands.

```
export PATH ${INFORMIXDIR}/bin:path_to_ODBC_driver_manager/bin:${PATH}

export LD_LIBRARY_PATH $INFORMIXDIR/lib:$INFORMIXDIR/lib/esql:path_to_ODBC_driver_
manager/lib:$LD_LIBRARY_PATH

export ODBCINI =/etc/odbc.ini  # depends which odbc.ini file contains your DSN
entry

isql -v superstores informix informix123
```

3.1.3 Configuring ODBC on Windows

This section describes how to configure ODBC on a Windows system including configuring your DSN. You can create a user DSN or a system DSN as both are supported.

The following process uses the Windows default ODBC client and more information is provided with the context-sensitive help and in the *IBM Informix ODBC Driver Programmer's Manual*.

1. Select **Start** then **Control Panel** then **Administrative Tools** then **Data Sources (ODBC)** to run the default ODBC client.

2. (Optional) To configure a system DSN, click the **System DSN** tab. All subsequent steps are the same to configure either DSN.

3. Click **Add**.

The Create New Data Source dialog box appears.

4. Under Create New Data Source, select the client that matches your Informix Dynamic Server version.

5. Click **Finish**. The Create a New Data Source wizard is displayed.

6. Supply the following:

   - **Name**: Can be of your choosing. In a Windows cluster, use one name across all nodes in the cluster.

   - **Description**: (Optional) Type a description of this data source.

   - **Server**: Select the Informix Dynamic Server. For Extract, select the `syscdcv1` database field. For Replicat, select the database field as the name of the target database. The DSN *must* be this database.

7. Click **Next**.

8. For login authentication, select **With Informix Dynamic Server authentication using a login ID and password entered by the user** for Oracle GoldenGate to use database credentials. Supply login information for the `informix` user. The `informix` user *must* be used by all Oracle GoldenGate processes.

9. Click **Next**.

10. If the default database is not set to the one that Oracle GoldenGate will connect to, click **Change the default database to**, and then select the correct name.

11. Set the other settings to use ANSI.

12. Click **Next**.

13. Leave the next page set to the defaults.

14. Click **Finish**.

15. Click **Test Data Source** to test the connection.

16. Close the confirmation dialog box and the Create a New Data Source dialog box.

17. Repeat this procedure on each Informix Dynamic Server source and target system.

3.2 Preparing Tables for Processing

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

3.2.1 Disabling Triggers and Cascade Constraints on the Target

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.

3.2.2 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. If a table does not have a defined row identifier, such as a primary key, Oracle GoldenGate constructs one from all columns that can be used in a SQL `WHERE` clause.

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

Unless a `KEYCOLS` clause is used in the `TABLE` or `MAP` statement (see Section 3.2.2.2), Oracle GoldenGate selects a row identifier to use in the following order of priority:

1. Primary key constraint column(s).

2. If no primary key exists in the table, then all of the non-LOB columns together are used by Oracle GoldenGate to form the primary key.

3.2.2.2 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 overrides any existing primary or unique key that Oracle GoldenGate finds. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.

3.2.3 Limiting Row Changes in Tables that Do Not Have a Key

If a target table has no primary key or unique key, duplicate rows can exist. It is possible for Oracle GoldenGate to update or delete too many rows in such a table, 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.

3.2.4 Improving Performance with Array Processing

You can improve the performance of Replicat by using the `BATCHSQL` parameter in the Replicat parameter file. `BATCHSQL` causes Replicat to use array processing instead of applying SQL statements one at a time. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.

3.3 Preparing the Logical Logs for Capture

This section provides instructions for sizing the transaction logs and enabling logging for the tables that you want to include in the Oracle GoldenGate configuration. Change data from the logged tables is passed to the Extract capture process.

3.3.1 Sizing the Logical Logs

The `LOGSIZE` and `LOGFILES` configuration parameters indirectly affect Oracle GoldenGate checkpoints because they specify the size and number of the logical-log files. They may need to be adjusted.

Informix archives the active log files on a rolling basis. For example, if there are 10 online log files, when the number-10 file is filled up, the number-1 log file gets archived and the log writer is then ready to write to a new online log.

The current log that is being processed by Oracle GoldenGate should never be the one that will be archive next. You can check the current LSN that is being processed by Oracle GoldenGate by issuing the `INFO EXTRACT` *group* command in GGSCI. Then, use the `onstat-1` command to check the online log status of the Informix server. Based on this information, you can define the number and the size of the online log files to prevent them from being archived before Extract is finished processing them. It is recommended that you configure the file size to `1048576` as follows:

```
onmode -wf LOGSIZE=1048576
```

3.3.2 Configuring Logging Properties

You must enable full-row logging for the tables that you want to include in the Oracle GoldenGate configuration. The logging must be enabled on the source system so that the log data is available to pass to the Extract process for capture. If using a bi-directional configuration, logging must be enabled on both systems in the configuration.

3.3.2.1 Enabling Full-Row Logging on Tables

Full-row logging must be enabled using the Oracle GoldenGate GGSCI interface:

1. Run GGSCI on the source system.

2. Issue the `DBLOGIN` command as the `informix` user. Use one of the following options, depending on your security configuration.

    ■ If using the Oracle GoldenGate credential store:

    ```
    DBLOGIN USERIDALIAS alias_of_informix_user
    ```

    ■ If not using the Oracle GoldenGate credential store:

    ```
    DBLOGIN SOURCEDB DSN_of_syscdc USERID informix PASSWORD informix_password
    algorithm ENCRYPTKEY {keyname | DEFAULT}]
    ```

    See *Reference for Oracle GoldenGate for Windows and UNIX* for more information about `DBLOGIN` syntax.

3. Issue the `ADD TRANDATA` command.

    ```
    ADD TRANDATA database.schema.table
    ```

    You can use a wildcard for the schema and table name components in the command input.

See *Reference for Oracle GoldenGate for Windows and UNIX* for more information about `ADD TRANDATA`.

3.4 Supporting the Capture of LOBs

To configure Extract to fetch a LOB, include the FETCHCOLS option in the TABLE statement that specifies the table where the LOB is stored. TABLE statements are used to specify the database objects that you want to capture for replication by Oracle GoldenGate. For more information about TABLE and other configuration parameters, see *Administering Oracle GoldenGate for Windows and UNIX* and *Reference for Oracle GoldenGate for Windows and UNIX*.

There is a known memory leak in the Informix ODBC driver that causes Replicat in classic Capture mode to error. Use the following steps to correct this issue:

1. Install the CSDK 4.10 FC4 into a separate directory.

2. Export LD_LIBRARY_PATH or the respective operating specific path for library search to this CSDK directory. For example:

   ```
   export LD_LIBRARY_PATH=CSDKInstallPath/lib/cli:CSDKInstallPath/lib/esql:$LD_
   LIBRARY_PATH
   ```

3. Run STOP MANAGER.

4. Run START MANAGER.

5. Run START REPLICAT.

4

Configuring Oracle GoldenGate for Capture

This chapter contains instructions for configuring the Oracle GoldenGate Extract (capture) process to capture data from an Informix source. The Extract process captures change data from the Informix transaction log and processes that data for replication before writing it to a trail for temporary storage. For more information about the Extract process, see *Administering Oracle GoldenGate for Windows and UNIX*.

This chapter includes the following sections:

- Section 4.1, "Prerequisites for Configuring Capture"
- Section 4.2, "What to Expect from these Instructions"
- Section 4.3, "Configuring the Primary Extract"
- Section 4.4, "Configuring the Data Pump Extract"
- Section 4.5, "Next Steps"

4.1 Prerequisites for Configuring Capture

The guidelines in the following sections should be satisfied before configuring Extract.

1. Section 1, "System Requirements and Preinstallation Instructions"
2. Section 2, "Installing Oracle GoldenGate"
3. Section 3, "Preparing the System for Oracle GoldenGate"
4. Create the Oracle GoldenGate instance on the source system by configuring the Manager process. See *Administering Oracle GoldenGate for Windows and UNIX*.
5. Additionally, review the guidelines in *Administering Oracle GoldenGate for Windows and UNIX*.

4.2 What to Expect from these Instructions

These instructions show you how to configure the following:

- A basic Extract parameter (configuration) file for the primary Extract, which captures transaction data from the Informix transaction log.
- A data-pump Extract, which propagates captured data that is stored locally in a trail from the source system to the target system.

Your business requirements probably will require a more complex topology, but this procedure forms a basis for the rest of your configuration steps. For more information about Oracle GoldenGate architecture, see *Administering Oracle GoldenGate for Windows and UNIX*.

By performing these steps, you can:

- get the basic configuration files established.

- build upon them later by adding more parameters as you make decisions about features or requirements that apply to your environment.

- use copies of them to make the creation of additional parameter files faster than starting from scratch.

4.3 Configuring the Primary Extract

These steps configure Extract to capture transaction data from the Oracle GoldenGate API.

1. In GGSCI on the source system, create the Extract parameter file.

   ```
   EDIT PARAMS name
   ```

   Where: *name* is the name of the primary Extract.

2. Enter the Extract parameters in the order shown, starting a new line for each parameter statement. See Table 4–1 for more information and parameter descriptions.

   Basic parameters for the primary Extract

   ```
   EXTRACT finance
   SOURCEDB syscdcv1 USERIDALIAS tiger1
   LOGALLSUPCOLS
   ENCRYPTTRAIL AES192
   EXTTRAIL /ggs/dirdat/lt
   TABLE mydatabase.*.*;
   ```

Table 4–1   Basic Parameters for Primary Extract in Classic Capture Mode

Parameter	Description
EXTRACT *group*	*group* is the name of the Extract group. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.
SOURCEDB *DSN* USERIDALIAS *alias*	Specifies the ODBC data source (DSN) of the source database and the alias of the database login credential of the user that is assigned to Extract. The DSN must point to the syscdcv1 database. The USERIDALIAS credential must exist in the Oracle GoldenGate credential store. It is recommended that you use the credential store as a best practice. For more information about security options, see *Administering Oracle GoldenGate for Windows and UNIX*.
	Note: This specification is required by Oracle GoldenGate. When specified here, you do not need to specify the login credentials in the odbc.ini file (thus avoiding a security risk).
LOGALLSUPCOLS	Writes all supplementally logged columns to the trail, including those required for conflict detection and resolution. (Scheduling columns are primary key, unique index, and foreign key columns.) You configure the database to log these columns with GGSCI commands. See Section 3.3.2, "Configuring Logging Properties."

Table 4–1 (Cont.) Basic Parameters for Primary Extract in Classic Capture Mode

Parameter	Description
`ENCRYPTTRAIL algorithm`	Encrypts the local trail. For more information about Oracle GoldenGate trail encryption options, see *Administering Oracle GoldenGate for Windows and UNIX*.
`EXTTRAIL pathname [FORMAT RELEASE release]`	Specifies the path name of the local trail to which the primary Extract writes captured data.
	If a version 12.x Extract is sending data to a downstream process that is an 11*g* or earlier version, the `FORMAT` option must be used to write the object names to the trail in the proper format. Earlier versions of Oracle GoldenGate do not support three-part naming conventions (`database.schema.table`). The `FORMAT` option directs Extract to write the two-part names to the trail.
	For more information, see *Reference for Oracle GoldenGate for Windows and UNIX.*
`TABLE database.schema.object [, FETCHCOLS (LOB_column[, LOB_column[, ...])]` `;`	Specifies the database object for which to capture data.
	■ `TABLE` s is a required keyword.
	■ `database` specifies the Informix database that contains the objects to be captured. No wildcard is allowed for this component.
	■ `schema` is the schema name or a wildcarded set of schemas.
	■ `object` is the table or sequence name, or a wildcarded set of those objects.
	■ `FETCHCOLS` specifies one or more LOB columns to fetch for the specified table.
	See *Administering Oracle GoldenGate for Windows and UNIX* for information about how to specify object names with and without wildcards.
	Terminate the parameter statement with a semi-colon (`;`).
	To exclude tables from a wildcard specification, use the `TABLEEXCLUDE` parameter. See *Reference for Oracle GoldenGate for Windows and UNIX* for more information about usage and syntax.
	For more information and for additional `TABLE` options that control data filtering, mapping, and manipulation, see *Reference for Oracle GoldenGate for Windows and UNIX*.

3. Enter any optional Extract parameters that are recommended for your configuration. You can edit this file at any point before starting processing by using the `EDIT PARAMS` command in GGSCI. For more information about optional parameters that control error handling, performance, and other aspects of processing, see *Reference for Oracle GoldenGate for Windows and UNIX*. For broader task-based information about how to configure Oracle GoldenGate, see *Administering Oracle GoldenGate for Windows and UNIX*

4. Save and close the file.

4.4 Configuring the Data Pump Extract

These steps configure the data pump that reads a local trail and sends the data across the network to a remote trail. For more information about data pumps, see *Administering Oracle GoldenGate for Windows and UNIX*.

1. In GGSCI on the source system, create the data-pump parameter file.

   ```
   EDIT PARAMS name
   ```

Where *name* is the name of the data pump Extract.

2. Enter the data-pump parameters in the order shown, starting a new line for each parameter statement. Your input variables will be different. See Table 4–2 for descriptions.

Basic parameters for the data-pump Extract group:

```
EXTRACT extpump
RMTHOST fin1, MGRPORT 7809 ENCRYPT AES192, KEYNAME securekey2
RMTTRAIL /ggs/dirdat/rt
TABLE mydatabase.*.*;
```

Table 4–2   Basic Parameters for a Data-pump Extract

Parameter	Description
`EXTRACT group`	*group* is the name of the data pump Extract. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX.*

`RMTHOST hostname,`
`MGRPORT portnumber,`
`[, ENCRYPT algorithm`
`KEYNAME keyname]`
	■ `RMTHOST` specifies the name or IP address of the target system.
	■ `MGRPORT` specifies the port number where Manager is running on the target.
	■ `ENCRYPT` specifies optional encryption of data across TCP/IP.
	
	For additional options and encryption details, see *Reference for Oracle GoldenGate for Windows and UNIX.*

`RMTTRAIL pathname`
`[FORMAT RELEASE`
`release]`
	Specifies the path name of the remote trail to which the data-pump Extract writes captured data
	
	If a version 12.x Extract is sending data to a downstream process that is an 11*g* or earlier version, the `FORMAT` option must be used to write the object names to the trail in the proper format. Earlier versions of Oracle GoldenGate do not support three-part naming conventions (`database.schema.table`). The `FORMAT` option directs Extract to write the two-part names to the trail.
	
	For more information, see *Reference for Oracle GoldenGate for Windows and UNIX.*

`TABLE` `database.schema.object;`	Specifies the database object for which to capture data. ■ `TABLE` s is a required keyword. ■ *database* specifies the same Informix database that is specified for the primary Extract. ■ *schema* is the schema name or a wildcarded set of schemas. ■ *object* is the table or sequence name, or a wildcarded set of those objects. See *Administering Oracle GoldenGate for Windows and UNIX* for information about how to specify object names with and without wildcards. Terminate the parameter statement with a semi-colon (`;`). To exclude tables from a wildcard specification, use the `TABLEEXCLUDE` parameter. See *Reference for Oracle GoldenGate for Windows and UNIX* for more information about usage and syntax. For more information and for additional `TABLE` options that control data filtering, mapping, and manipulation, see *Reference for Oracle GoldenGate for Windows and UNIX.*

3. Enter any optional Extract parameters that are recommended for your configuration. You can edit this file at any point before starting processing by using the `EDIT PARAMS` command in GGSCI. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX.*

4. Save and close the file.

4.5 Next Steps

To configure the Replicat (apply) process, see Chapter 5, "Configuring Oracle GoldenGate for Apply."

5

Configuring Oracle GoldenGate for Apply

This chapter contains instructions for configuring the Oracle GoldenGate Replicat (apply) process to apply data to an Informix target. The Replicat process reads change data from the trail on the target system and applies that data to the Informix target. For more information about the Replicat process, see *Administering Oracle GoldenGate for Windows and UNIX*.

This chapter includes the following sections:

- Section 5.1, "Prerequisites for Configuring Replicat"
- Section 5.2, "What to Expect from these Instructions"
- Section 5.3, "Creating a Checkpoint Table"
- Section 5.4, "Configuring Replicat"
- Section 5.5, "Advanced Configuration Options for Oracle GoldenGate"
- Section 5.6, "Next Steps"

5.1 Prerequisites for Configuring Replicat

The guidelines in the following sections should be satisfied before configuring Replicat.

- Section 1, "System Requirements and Preinstallation Instructions"
- Section 2, "Installing Oracle GoldenGate"
- Section 3, "Preparing the System for Oracle GoldenGate"
- Section 4, "Configuring Oracle GoldenGate for Capture"
- Create the Oracle GoldenGate instance on the target system by configuring the Manager process. See *Administering Oracle GoldenGate for Windows and UNIX*.
- Additionally, review the guidelines in *Administering Oracle GoldenGate for Windows and UNIX*.
- When using coordinated apply, the target tables (DDL) must have primary key columns defined.

5.2 What to Expect from these Instructions

These instructions show you how to configure a basic Replicat parameter (configuration) file. Your business requirements probably will require a more complex topology, but this procedure forms a basis for the rest of your configuration steps.

By performing these steps, you can:

- get the basic configuration file established.

- build upon it later by adding more parameters as you make decisions about features or requirements that apply to your environment.

- use copies of it to make the creation of additional Replicat parameter files faster than starting from scratch.

5.3 Creating a Checkpoint Table

Replicat maintains its recovery checkpoints in a table, known as the *checkpoint table*, that is stored in the target database. Checkpoints are written to the checkpoint table within the Replicat transaction. Because a checkpoint either succeeds or fails with the transaction, Replicat ensures that a transaction is only applied once, even if there is a failure of the process or the database.

> **Note:** This procedure installs a default checkpoint table, which is sufficient in most cases. More than one checkpoint table can be used, such as to use a different one for each Replicat group. To use a non-default checkpoint table, which overrides the default table, use the CHECKPOINTTABLE option of ADD REPLICAT when you create Replicat process.

5.3.1 Adding the Checkpoint Table to the Target Database

1. From the Oracle GoldenGate directory on the target, run GGSCI.

2. Issue the DBLOGIN command as the informix user. Use one of the following options, depending on your security configuration.

   - If using the Oracle GoldenGate credential store:

     ```
     DBLOGIN USERIDALIAS alias_of_informix_user
     ```

   - If not using the Oracle GoldenGate credential store:

     ```
     DBLOGIN SOURCEDB DSN_of_syscdc USERID informix PASSWORD informix_password
     algorithm ENCRYPTKEY {keyname | DEFAULT}]
     ```

   See *Reference for Oracle GoldenGate for Windows and UNIX* for more information about DBLOGIN syntax.

3. In GGSCI, create the checkpoint table in a schema of your choice (ideally dedicated to Oracle GoldenGate).

   ```
   ADD CHECKPOINTTABLE database.schema.table
   ```

   You can use a wildcard for any or all of the name components in the command input.

5.3.2 Specifying the Checkpoint Table in the Oracle GoldenGate Configuration

To specify the checkpoint table in the Oracle GoldenGate configuration:

1. Create a GLOBALS file (or edit the existing one).

   ```
   EDIT PARAMS ./GLOBALS
   ```

> **Note:** `EDIT PARAMS` creates a simple text file. When you save the file after `EDIT PARAMS`, it is saved with the name `GLOBALS` in upper case, without a file extension. It must remain as such, and the file must remain in the root Oracle GoldenGate directory.

2. In the `GLOBALS` file, enter the `CHECKPOINTTABLE` parameter.

    ```
    CHECKPOINTTABLE database.schema.table
    ```

3. Save and close the `GLOBALS` file.

5.4 Configuring Replicat

These steps configure the Replicat process. For more advanced mapping options, see *Administering Oracle GoldenGate for Windows and UNIX*.

> **Note:** This configuration is a basic Replicat configuration to get you started. You may want to enhance this configuration by adding parameters that configure Replicat in coordinated mode. The coordinated mode uses parallel apply threads to enhance apply performance. For more information, see *Administering Oracle GoldenGate for Windows and UNIX*.

1. In GGSCI on the target system, create the Replicat parameter file.

    ```
    EDIT PARAMS name
    ```

    Where: `name` is the name of the Replicat group.

2. Enter the Replicat parameters in the order shown, starting a new line for each parameter statement. See Table 5–1 for descriptions.

    Basic parameters for the Replicat group:

    ```
    REPLICAT financer
    TARGETDB informix USERIDALIAS tiger2
    ASSUMETARGETDEFS
    MAP mydatabase.*.*, TARGET mydatabase2.*.*;
    ```

Table 5–1 Basic Parameters for Replicat

Parameter	Description
REPLICAT *group*	*group* is the name of the Replicat group.
TARGETDB *DSN* USERIDALIAS *alias*	Specifies the ODBC data source (DSN) of the target database and the alias of the database login credential of the user that is assigned to Replicat. This credential must exist in the Oracle GoldenGate credential store. It is recommended that you use the credential store as a best practice. For more information about security options, see *Administering Oracle GoldenGate for Windows and UNIX.*
	Note: This specification is required by Oracle GoldenGate. When specified here, you do not need to specify the login credentials in the odbc.ini file (thus avoiding a security risk).
ASSUMETARGETDEFS	Specifies how to interpret data definitions. ASSUMETARGETDEFS assumes the source and target tables have identical definitions, including semantics. (This procedure assume identical definitions.)
	Use the alternative SOURCEDEFS if the source and target tables have different definitions, and create a source data-definitions file with DEFGEN. For more information about data definitions, see *Administering Oracle GoldenGate for Windows and UNIX.*
MAP *database.schema.object*, TARGET *database.schema.object*;	Specifies the relationship between a source table or sequence, or multiple objects, and the corresponding target object or objects.
	■ MAP is a required keyword. The MAP clause specifies the source object or objects that are to be mapped to target object(s).
	■ TARGET is a required keyword. The TARGET clause specifies the target object or objects to which the source object(s) are mapped.
	■ *database* is the name of a database. Wildcards are not allowed for this component.
	■ *schema* is the schema name or a wildcarded set of schemas.
	■ *object* is the name of a table or sequence, or a wildcarded set of objects.
	Terminate this parameter statement with a semi-colon (;).
	For cross database replication, Oracle Extract to Informix Replicat, the MAP table name on the source can be a two-part name. For example:
	`MAP schema.tabname, TARGET mydatabase.schema.tabname;`
	For a like-like database replication, Informix Extract to Informix Replicat, the MAP parameter should be used as follows:
	`MAP database.schema.tabname, TARGET mydatabase.schema.tabname;`
	Note: If your data contains 0x00, then you must use the BYTE data type because Informix interprets zero values as terminators.
	For three-part table names used by SQLEXCE QUERY queries, you must define the names with targetdb:informix.set_tcustmer_sqltrc as in the following example:
	`map sourcedb.informix.set_tcustmer_sqltrc, target targetdb.informix."set_tcustmer_sqltrc",`
	To exclude objects from a wildcard specification, use the MAPEXCLUDE parameter.
	For more information and for additional options that control data filtering, mapping, and manipulation, see MAP in *Reference for Oracle GoldenGate for Windows and UNIX.*

3. Enter any optional Replicat parameters that are recommended for your configuration. You can edit this file at any point before starting processing by using the EDIT PARAMS command in GGSCI. For more information about optional

parameters that control error handling, performance, and other aspects of processing, see *Reference for Oracle GoldenGate for Windows and UNIX*. For broader task-based information about how to configure Oracle GoldenGate, see *Administering Oracle GoldenGate for Windows and UNIX*.

4. Save and close the file.

5.5 Advanced Configuration Options for Oracle GoldenGate

You may need to configure Oracle GoldenGate with advanced options to suit your business needs. See the following:

- For additional configuration guidelines to achieve specific replication topologies, see *Administering Oracle GoldenGate for Windows and UNIX*. This guide includes instructions for the following configurations:

  - Using Oracle GoldenGate for live reporting

  - Using Oracle GoldenGate for real-time data distribution

  - Configuring Oracle GoldenGate for real-time data warehousing

  - Using Oracle GoldenGate to maintain a live standby database

  - Using Oracle GoldenGate for active-active high availability

  That guide also contains information about:

  - Oracle GoldenGate architecture

  - Oracle GoldenGate commands

  - Oracle GoldenGate initial load methods

  - Configuring security

  - Using customization features

  - Configuring data filtering and manipulation

5.6 Next Steps

Once you have created a basic parameter file for Replicat, see Chapter 6, "Adding Process Groups."

6

Adding Process Groups

This chapter contains instructions for creating Oracle GoldenGate process groups, collectively known as the "change-synchronization" processes. At minimum, you will create one primary Extract, one data pump, and one Replicat process group. You configured these groups in the following chapters:

Chapter 4, "Configuring Oracle GoldenGate for Capture"

Chapter 5, "Configuring Oracle GoldenGate for Apply"

This chapter guides you through the process of actually creating these processes. Each Extract and Replicat process is known as a *process group*. For more information about Oracle GoldenGate process groups, see *Administering Oracle GoldenGate for Windows and UNIX*.

This chapter includes the following sections:

- Section 6.1, "Prerequisites to These Procedures"
- Section 6.2, "Adding the Primary Extract"
- Section 6.3, "Adding the Local Trail"
- Section 6.4, "Add the Data Pump Extract Group"
- Section 6.5, "Add the Remote Trail"
- Section 6.6, "Add the Replicat Group"
- Section 6.7, "Next Steps"

6.1 Prerequisites to These Procedures

This chapter assumes you have installed Oracle GoldenGate, understand the different processing options available to you, and have performed the following prerequisite configuration steps before proceeding to configure Oracle GoldenGate process groups:

Chapter 1, "System Requirements and Preinstallation Instructions"

Chapter 2, "Installing Oracle GoldenGate"

Chapter 3, "Preparing the System for Oracle GoldenGate"

Chapter 4, "Configuring Oracle GoldenGate for Capture"

Chapter 5, "Configuring Oracle GoldenGate for Apply"

6.2 Adding the Primary Extract

These steps add the primary Extract that captures change data.

1. Run GGSCI.

2. Issue the `ADD EXTRACT` command to add the primary Extract group.

   ```
   ADD EXTRACT group, VAM, BEGIN NOW
   ```

   Where:

   - *group* is the name of the Extract group.

   - `VAM` specifies the VAM (Vendor Access Module) as the data source. For more information about the VAM, see *Administering Oracle GoldenGate for Windows and UNIX*.

   - `BEGIN NOW` starts at the end of the log file. Positioning at a particular LSN is also possible when using BEGIN with the LSN option, such as the following:

     ```
     ADD EXTRACT group, BEGIN LSN 139:0xc68018
     ```

     > **Note:** Oracle GoldenGate Extract processes read transaction log records from the database. If Extract is positioned to start at any valid LSN and the database is idle, the transaction records are not pushed to the extract until the next database checkpoint occurs. You can force your Informix database to checkpoint using the `onmode -c` command.

   Example 6–1   Add Extract group named "finance"

   ```
   ADD EXTRACT finance, VAM, BEGIN NOW
   ```
   This command initializes the position in the Informix logical log to capture the record. When the first DML is processed, the record is contained in the logical log. Each record in the log is associated with a position, which is called LSN (Log Sequence Number). Only when initialization is complete will it be able to honor the positioning based on the LSN. The very first time you add an Extract, you *must* ensure that the has initialization completed (the duration depends on number of tables in your Extract parameter file) using the `INFO EXTRACT` command.

   > **Note:** For detailed syntax information, see *Reference for Oracle GoldenGate for Windows and UNIX*.

3. Issue the `INFO EXTRACT` command.

4. Ensure that the LSN number displayed matches the LSN displayed as the first record processed in the Extract report file. For example, if the LSN number returned by `INFO EXTRACT` is `LSN: 892:0X1235018`, then the message in the report file must be `Position of first record processed LSN: 892:0X1235018, Apr 16, 2014 2:56:58 AM`.

   > **Note:** *Do not* issue a stop or kill command until you have verified the LSN positioning.

   When the initial Extract log positioning is complete, you can issue a stop or kill command *though not before*; doing so before results in any capture restart always starting from the EOF position of the database logs.

6.3 Adding the Local Trail

This adds the local trail to which the primary Extract writes captured data. For more information about Oracle GoldenGate and the trail naming conventions, see *Administering Oracle GoldenGate for Windows and UNIX.*

In GGSCI on the source system, issue the `ADD EXTTRAIL` command:

```
ADD EXTTRAIL path, EXTRACT group
```

Where:

- `EXTTRAIL` specifies that the trail is to be created on the local system.
- `path` is the relative or fully qualified name of the trail, including the two-character name.
- `EXTRACT group` is the name of the primary Extract group.

Example 6–2

```
ADD EXTTRAIL /ggs/dirdat/lt, EXTRACT finance
```

> **Note:**   Oracle GoldenGate creates this trail automatically during processing.

6.4 Add the Data Pump Extract Group

This adds the data pump that reads the local trail and sends the data to the target.

In GGSCI on the source system, issue the `ADD EXTRACT` command.

```
ADD EXTRACT group, EXTTRAILSOURCE trail
```

Where:

- `group` is the name of the Extract group.
- `EXTTRAILSOURCE trail` is the relative or fully qualified name of the local trail.

Example 6–3

```
ADD EXTRACT financep, EXTTRAILSOURCE /ggs/dirdat/lt
```

6.5 Add the Remote Trail

These steps add the remote trail. Although it is read by Replicat, this trail must be associated with the data pump, so it must be added on the source system, not the target. For more information about Oracle GoldenGate and the trail naming conventions, see *Administering Oracle GoldenGate for Windows and UNIX.*

In GGSCI on the source system, issue the following command:

```
ADD RMTTRAIL path, EXTRACT group
```

Where:

- `RMTTRAIL` specifies that the trail is to be created on the target system.

- *path* is the relative or fully qualified name of the trail, including the two-character name.

- `EXTRACT` *group* is the name of the data-pump Extract group.

Example 6–4

```
ADD RMTTRAIL /ggs/dirdat/rt, EXTRACT financep
```

> **Note:**   Oracle GoldenGate creates this trail automatically during processing.

6.6 Add the Replicat Group

These steps add the Replicat group that reads the remote trail and applies the data changes to the target Informix database.

1. Run GGSCI on the target system.

2. Issue the `ADD REPLICAT` command with the following syntax.

```
ADD REPLICAT group, COORDINATED [MAXTHREADS number], EXTTRAIL path
```

Where:

- *group* is the name of the Replicat group.

- `COORDINATED [MAXTHREADS` *number*`]` specifies that this group is a coordinated Replicat group. Use this option only if you configured the parameter file as a coordinated Replicat group in Chapter 5, "Configuring Oracle GoldenGate for Apply."

- `EXTTRAIL` *path* is the relative or fully qualified name of the remote trail, including the two-character name.

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

Example 6–5   Add a Replicat group named "financer"

```
ADD REPLICAT financer, EXTTRAIL /ggs/dirdat/rt
```

6.7 Next Steps

The next step, from a simple standpoint, is to instantiate replication processing by performing an initial synchronization of the data and starting the Oracle GoldenGate processes. However, most business requirements extend well beyond the simple configuration that is provided in this manual. Oracle GoldenGate is very flexible, and you should understand the options available to you before you start the processes. It is strongly recommended that you review *Administering Oracle GoldenGate for Windows and UNIX* to learn more about the Oracle GoldenGate architecture and the many additional configuration options available to you to expand and customize Oracle GoldenGate to meet your specific requirements.

7

Uninstalling Oracle GoldenGate

This chapter contains the procedures for uninstalling Oracle GoldenGate for Informix. It 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.

This chapter includes the following sections:

- Section 7.1, "Stopping Processes"
- Section 7.2, "Disabling Logging"
- Section 7.3, "Uninstalling Oracle GoldenGate"

7.1 Stopping Processes

This procedure stops the Extract and Replication processes. Leave Manager running until directed to stop it.

On all Systems:

1. Run the command shell.

2. 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.

   ```
   STOP ER *
   ```

6. Stop the Manager process.

   ```
   STOP MANAGER
   ```

7.2 Disabling Logging

Optionally, you can disable full-row logging for the tables that were being captured with Oracle GoldenGate.

1. Run GGSCI.

2. Issue the DBLOGIN command as the informix user. Use one of the following options, depending on your security configuration.

   - If using the Oracle GoldenGate credential store:

```
DBLOGIN USERIDALIAS alias_of_informix_user
```

- If not using the Oracle GoldenGate credential store:

```
DBLOGIN SOURCEDB DSN_of_syscdc USERID informix PASSWORD informix_password
algorithm ENCRYPTKEY {keyname | DEFAULT}]
```

See *Reference for Oracle GoldenGate for Windows and UNIX* for more information about DBLOGIN syntax.

3. To determine which tables have full-row logging enabled, issue the INFO TRANDATA command.

```
INFO TRANDATA database.schema.table
```

You can use a wildcard for any or all of the name components in the command input.

4. Issue the DELETE TRANDATA command.

```
DELETE TRANDATA database.schema.table
```

You can use a wildcard for the schema and name components in the command input.

By default, DELETE TRANDATA issues the following command to the database.

```
EXECUTE FUNCTION SYSCDC_database_name:informix.cdc_set_
fullrowlogging('database:schema.table',0)
```

The default SYSCDC database name is syscdcv1.

5. (Optional) Drop the SYSCDC database that you created for Oracle GoldenGate.

7.3 Uninstalling Oracle GoldenGate

Follow these instructions to remove the Oracle GoldenGate environment from the system.

7.3.1 Removing Oracle GoldenGate Windows Components

(Valid for Windows installations) This procedure does the following: removes Oracle GoldenGate as a Windows cluster resource from a source or target Windows system, stops Oracle GoldenGate events from being reported to the Windows Event Manager, and removes the Manager service. Perform these steps on source and target systems.

1. 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. Click **Start** then **Run**, and then type cmd in the Run dialog box to open the command console.

3. Change directories to the Oracle GoldenGate installation directory.

4. Run the INSTALL utility with the following syntax.

```
install deleteevents deleteservice
```

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

7.3.2 Removing the Oracle GoldenGate Files

Perform these steps on all systems to remove the Oracle GoldenGate installation directory.

1. In GGSCI, verify that all processes are stopped. Stop any that are running.

    ```
    STATUS MANAGER
    STATUS ER *
    STOP MANAGER
    STOP ER *
    ```

2. Exit GGSCI.

    ```
    EXIT
    ```

3. Remove the Oracle GoldenGate 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   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.
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.

Table A–1 (Cont.) Programs and Utilities

Program	Description
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.
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.

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 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 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.

Table A–2   (Cont.) Subdirectories

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.
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`

Table A–2   (Cont.) Subdirectories

Directory	Description
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.

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 operating system platform.

Table A–3   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` 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_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_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)
`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)

Table A–3 (Cont.) Other Files

Component	Description
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 utility.
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.
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.

Table A–3 (Cont.) Other Files

Component	Description
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)
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 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 A–4   Checkpoint table definitions

Column	Definition
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.
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.