## Contents

<table>
<thead>
<tr>
<th>Chapter 1</th>
<th>System requirements and preinstallation instructions</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview of Oracle GoldenGate for SQL Server</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Supported editions of SQL Server</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Supported Platforms</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Operating system requirements</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Memory requirements</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Disk requirements</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Temporary disk requirements</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Operating system privileges</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Console</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Other programs</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>SQL Server requirements</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Instance configuration</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Database configuration</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Database connection</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Database connectivity drivers</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Database user for Oracle GoldenGate processes</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Supported SQL Server data types</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Non-supported SQL Server data types</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Supported objects and operations for SQL Server</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Non-supported objects and operations for SQL Server</td>
<td>8</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 2</th>
<th>Installing Oracle GoldenGate</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Installation overview</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Downloading Oracle GoldenGate</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>Preparing to install Oracle GoldenGate within a cluster</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Deciding where to install Oracle GoldenGate binaries and files in the cluster</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Installing Oracle GoldenGate on Windows</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>Installing Oracle GoldenGate into a Windows Cluster</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>Installing the Oracle GoldenGate files</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>Specifying a custom Manager name</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>Installing Manager as a Windows service</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>Integrating Oracle GoldenGate into a cluster</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>General requirements in a cluster</td>
<td>14</td>
<td></td>
</tr>
</tbody>
</table>
Adding Oracle GoldenGate as a Windows cluster resource ........................................... 14

Chapter 3 Preparing the system for Oracle GoldenGate ...................................................... 15
Choosing Replicat database connection options ............................................................ 15
Using ODBC or default OLE DB ....................................................................................... 15
Using OLE DB with NOT FOR REPLICATION ................................................................... 16
Configuring an ODBC connection .................................................................................... 17
Preparing tables for processing ....................................................................................... 18
Disabling triggers and cascade constraints on the target ................................................. 18
Assigning row identifiers ................................................................................................. 18
Limiting row changes in tables that do not have a key ................................................... 19
Improving IDENTITY replication with array processing ............................................... 19

Chapter 4 Preparing for Oracle GoldenGate capture ....................................................... 21
Setting the database to full recovery model ..................................................................... 21
Backing up the transaction log ......................................................................................... 21
Retaining the log backups ............................................................................................... 22
Enabling supplemental logging ....................................................................................... 22
Managing the secondary truncation point ..................................................................... 23
Oracle GoldenGate manages the secondary truncation point ....................................... 24
SQL Server manages the secondary truncation point ..................................................... 25
Making a full database backup before you start Oracle GoldenGate ............................... 25

Chapter 5 Uninstalling Oracle GoldenGate ..................................................................... 27
Disabling supplemental logging ..................................................................................... 27
Removing Oracle GoldenGate from Windows Cluster .................................................. 28
Uninstalling Oracle GoldenGate from Windows (non-cluster) ....................................... 29

Appendix 1 Configuring the replication components (SQL Server 2005 pre-CU6 for SP2) ................................................................. 30
What the replication components do for Oracle GoldenGate ....................................... 30
Installing and configuring the SQL Server 2005 replication components ..................... 30
Configuring and cleaning up the distribution database ................................................. 31

Appendix 2 Oracle GoldenGate installed components .................................................... 33
Oracle GoldenGate Programs and Utilities ..................................................................... 33
Oracle GoldenGate subdirectories .................................................................................. 34
Other Oracle GoldenGate files ....................................................................................... 37
Oracle GoldenGate checkpoint table .............................................................................. 41

Index ................................................................................................................................. 43
CHAPTER 1
System requirements and preinstallation instructions

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

Overview of Oracle GoldenGate for SQL Server

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

Supported editions of SQL Server

- SQL Server 2005: Enterprise and Standard Edition
- SQL Server 2008: Enterprise (capture or delivery) or Standard Edition (delivery only)
- SQL Server 2008 R2: Enterprise (capture or delivery) or Standard Edition (delivery only)

Supported Platforms

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

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

Operating system requirements

Memory requirements

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

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

The actual amount of physical memory that is used by any Oracle GoldenGate process is controlled by the operating system, not the Oracle GoldenGate program. The Oracle GoldenGate cache manager takes advantage of the memory management functions of the operating system to ensure that Oracle GoldenGate processes work in a sustained and efficient manner. For more information about evaluating Oracle GoldenGate memory requirements, see the CACHEMGR parameter in the Oracle GoldenGate Windows and UNIX Reference Guide.

**Disk requirements**

Assign the following free disk space:

- 50-150 MB, depending on the database and platform. This includes space for the compressed download file and space for the uncompressed files. You can delete the download file after the installation is complete.
- 40 MB for the working directories and binaries for each instance of Oracle GoldenGate that you are installing on the system. For example, to install two builds of Oracle GoldenGate into two separate directories, allocate 80 MB of space.
- To install Oracle GoldenGate into a cluster environment, install the Oracle GoldenGate binaries and files as the Oracle user on a shared file system that is available to all cluster nodes. See “Preparing to install Oracle GoldenGate within a cluster” on page 28 for more information.
- An additional 1 GB of disk space on any system that hosts Oracle GoldenGate trails, which are files that contain the working data. You may need more or less than this amount, because the space that is consumed by the trails depends on the volume of data that will be processed. See the guidelines for sizing trails in the Oracle GoldenGate Administration Guide.

**Temporary disk requirements**

By default, Oracle GoldenGate maintains data that it swaps to disk in the dirtmp subdirectory 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 to this directory with the CACHEDIRECTORY option of the CACHEMGR parameter.

**Network**

- Configure the system to use TCP/IP services, including DNS. Oracle GoldenGate supports IPv4 and IPv6 and can operate in a system that supports one or both of these protocols.
- Configure the network with the host names or IP addresses of all systems that will be hosting Oracle GoldenGate processes and to which Oracle GoldenGate will be connecting. Host names are easier to use.
● Oracle GoldenGate requires some unreserved and unrestricted TCP/IP ports, the number of which depends on the number and types of processes in your configuration. See the Oracle GoldenGate Windows and UNIX Administrator’s Guide for details on how to configure the Manager process to handle the required ports.

● Keep a record of the ports that you assigned to Oracle GoldenGate. You will specify them with parameters when configuring the Manager process.

● Configure your firewalls to accept connections through the Oracle GoldenGate ports.

**Operating system privileges**

**Manager:** The Manager process can run as a Windows service, or it can run interactively as the current user. The Manager process requires:

● 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 (on all nodes in a cluster).

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

**Extract and Replicat:** See “Database user for Oracle GoldenGate processes” on page 5.

**Console**

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

```
chcp <code page number>
```

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

**Other programs**

● Before installing Oracle GoldenGate on a Windows system, install and configure the Microsoft Visual C++ 2005 SP1 Redistributable Package. **Make certain it is the SP1 version of this package, and make certain to get the correct bit version for your server.** This package installs runtime components of Visual C++ Libraries. For more information, and to download this package, go to http://www.microsoft.com.

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

Instance configuration

- For SQL Server 2008: the following must be true:
  - To capture from SQL Server 2008, the instance must be the Enterprise Edition of SQL Server.
  - Change Data Capture (CDC) must be enabled for Oracle GoldenGate and will be enabled by Oracle GoldenGate by means of the `ADD TRANDATA` command. See “Enabling supplemental logging” on page 22.

- To capture from a SQL Server 2005 database, you can do one of the following:
  - Run Oracle GoldenGate replication without having to install the SQL Server 2005 replication components. This requires the Microsoft Cumulative Update Package 6 (CU6) or greater for SQL Server 2005 Service Pack 2, or install Service Pack 3 or greater.
  - Run Oracle GoldenGate in conjunction with the SQL Server 2005 replication components. This configuration can be used whether or not Cumulative Update Package 6 (CU6 for SP2) or greater is installed, but is required if that upgrade is not installed. Oracle GoldenGate will use this configuration automatically if CU6 for SP2 or greater is not detected. To use this configuration, all tables that will be replicated must have a declared primary key. See “Configuring the replication components (SQL Server 2005 pre-CU6 for SP2)” on page 30 for more information.

- The SQL Server server name cannot be `NULL`.

Database configuration

- A SQL Server source database must be set to use the full recovery model.
- Oracle GoldenGate does not support system databases.
- After the source database is set to full recovery, a full database backup must be taken. This backup could be one that was already done, prior to the installation of Oracle GoldenGate, for a database that was previously using the full or bulk-logged recovery model. If you need to make a backup, see “Making a full database backup before you start Oracle GoldenGate” on page 25.
- The log chain on the source database must not be broken between the time of the last full database backup and the time when Oracle GoldenGate is installed. (The log chain is broken if the log was backed up with the “no_log” or “truncate_only” options, or if the recovery model was set to “simple” at any time after the initial full database backup was completed.) In addition, the log chain must not be broken after Oracle GoldenGate is installed. For more information, see the Microsoft SQL Server documentation on “Log Chains.”

Database connection

Oracle GoldenGate uses ODBC and/or OLE DB to connect to a database:

- **ODBC**: The Extract process uses ODBC (Open Database Connectivity) to connect to a source SQL Server database. The Replicat process uses ODBC to connect to a target SQL Server database to obtain metadata, but can optionally use it for its delivery transactions as well. ODBC must be properly configured. For more information, see “Configuring an ODBC connection” on page 17.
● **OLE DB**: By default, the Replicat process uses OLE DB to connect to a target SQL Server database to perform DML operations. (Thus there are always least two Replicat connections: ODBC for metadata and OLE DB for data). For more information about Replicat connection options, see page 15.

### Database connectivity drivers

The following drivers are required:

- SQL Server 2005 ODBC/OLE DB: SQL Native Client driver
- SQL Server 2008 ODBC/OLE DB: SQL Server Native Client 10.0 driver

### Database user for Oracle GoldenGate processes

The Oracle GoldenGate processes can use either Windows Authentication or SQL Server Authentication to connect to a database.

- To use Windows authentication, the Extract and Replicat processes inherit the login credentials of the Manager process, as identified by the Log On account specified in the Properties of the Manager service. This account must have the privileges listed in Table 1 on the source and target systems.
- To use SQL Server authentication, create a dedicated SQL Server login for Extract and Replicat and assign the privileges listed in Table 2. If using SQL Server authentication, you will need to specify the user and password with the USERID parameter (including the PASSWORD option) in the Extract or Replicat parameter file.

| Table 1 Required SQL Server privileges for Manager when using Windows authentication |
|-----------------------------------------------|-----------------------------------------------|-----------------------------------------------|
| Oracle GoldenGate Process | Manager privileges if using Local System account | Manager privileges if using local or domain account |
| Extract (source system) | BUILTIN\Administrators or NT AUTHORITY\SYSTEM account must be a member of the SQL Server fixed server role sysadmin. | Account must be a member of the SQL Server fixed server role sysadmin. |
| Replicat (target system) | BUILTIN\Administrators or NT AUTHORITY\SYSTEM account must be at least a member of the db_owner fixed database role of the target database. | Account must be at least a member of the db_owner fixed database role of the target database. |

| Table 2 Required SQL Server privileges for Extract and Replicat when using SQL Server authentication |
|-----------------------------------------------|-----------------------------------------------|
| Extract login | Replicat login |
| Member of the SQL Server fixed server role sysadmin. | At least a member of the db_owner fixed database role of the target database. |
Supported SQL Server data types

Oracle GoldenGate supports most SQL Server 2005 and 2008 data types except those listed under “Non-supported SQL Server data types”.

Limitations of support

- When the size of a large object exceeds 4K, Oracle GoldenGate stores the data in segments within the Oracle GoldenGate trail. The first 4K is stored in the base segment, and the rest is stored in a series of 2K segments. Oracle GoldenGate does not support the filtering, column mapping, or manipulation of large objects of this size. Full Oracle GoldenGate functionality can be used for objects that are 4K or smaller.
- Oracle GoldenGate treats XML data as a large object (LOB), as does SQL Server when the XML does not fit into a row. SQL Server 2008 extended XML enhancements (such as lax validation, DATETIME, union functionality) are not supported.
- A system-assigned TIMESTAMP column or a non-materialized computed column cannot be part of a key. A table containing a TIMESTAMP column must have a key, which can be a primary key, a unique constraint, or a substitute key specified with a KEYCOLS clause in the TABLE or MAP statement. See “Assigning row identifiers” on page 15.
- Oracle GoldenGate supports multi-byte character data types and multi-byte data stored in character columns. Multi-byte data is only supported in a like-to-like configuration. Transformation, filtering, and other types of manipulation are not supported for multi-byte character data.
- If data for TEXT, NTEXT, IMAGE, or VARCHAR(MAX), NVARCHAR(MAX) and VARBINARY(MAX) columns will exceed the SQL Server default size set for the max text repl size option, extend the size. Use sp_configure to view the current value of max text repl size.
- IDENTITY columns are supported as follows:
  - Fully in a uni-directional configuration.
  - Fully, or with limitations, in a bi-directional configuration, when the seed and increment values are configured properly, and depending on how the Replicat connection is defined. Full support, including support for IDENTITY ranges, requires OLE DB to be used with NOT FOR REPLICATION enabled for the IDENTITY columns and with Replicat operating as the replication user.
  - See Chapter 3 for more information.
- Oracle GoldenGate supports UDT and UDA data of up to 2 GB in size. All UDTs except SQL_Variant are supported. CLR (common language runtime), including SQL Server 2008 built-in CLR data types (such as geometry, geography and hierarchyid), are supported.
- The support of range and precision for floating-point numbers depends on the host machine. In general, the precision is accurate to 16 significant digits, but you should review the database documentation to determine the expected approximations. Oracle GoldenGate rounds or truncates values that exceed the supported precision.
- VARBINARY(MAX) column with the FILESTREAM attribute (introduced in SQL Server 2008) is supported up to a size of 4 GB. Extract uses standard Win32 file functions to read the FILESTREAM file.

Non-supported SQL Server data types

- SQL_Variant
Supported objects and operations for SQL Server

- Oracle GoldenGate supports the extraction and replication of DML operations on tables that contain rows of up to 512 KB in length. TEXT, NTEXT, IMAGE, VARBINARY, VARCHAR (MAX), and NVARCHAR(MAX) columns are supported in their full size.
- For SQL Server 2005, Oracle GoldenGate supports the maximum number of columns and the maximum column size per table that is supported by the database. For SQL Server 2008, Oracle GoldenGate supports the maximum sizes that are permitted for tables that are tracked by CDC.
- Oracle GoldenGate supports SQL Server 2008 tables that use data compression. This includes row compressed format in both ROW and PAGE mode.
- Oracle GoldenGate supports partitioned tables if the table has the same physical layout across all partitions.

Limitations on computed columns

- Oracle GoldenGate supports tables with non-persisted computed columns, but does not capture change data for these columns, because the database does not write it to the transaction log. To replicate data for non-persisted computed columns, you can use the FETCHCOLS or FETCHMODCOLS option of the TABLE parameter to fetch the column data from the table. Keep in mind that there can be discrepancies caused by differences in data values between when the column was changed in the database and when Extract fetches the data for the transaction record that is being processed.
- Replicat does not apply DML to any computed column, even if the data for that column is in the trail, because the database does not permit DML on that type of column. Data from a source persisted computed column, or from a fetched non-persisted column, can be applied to a target column that is not a computed column.
- In an initial load, all of the data is selected directly from the source tables, not the transaction log. Therefore, in an initial load, data values for all columns, including non-persisted computed columns, gets written to the trail or sent to the target, depending on the method that is being used. As when applying change data, however, Replicat does not apply initial load data to computed columns, because the database does not permit DML on that type of column.
- Oracle GoldenGate does not permit a non-persisted computed column to be used in a KEYCOLS clause in a TABLE or MAP statement.
- If a unique key includes a non-persisted computed column and Oracle GoldenGate must use that key, the non-persisted computed column will be ignored. This might affect data integrity if the remaining columns do not enforce uniqueness.
- If a unique index is defined on any non-persisted computed columns, it will not be used.
- If a unique key or index contains a non-persisted computed column and is the only unique identifier on a table, Oracle GoldenGate must use all of the columns as an identifier to find target rows. Because a non-persisted computed column cannot be used in this identifier, it is possible that Replicat could apply operations containing this identifier to the wrong target rows.
Non-supported objects and operations for SQL Server

- (SQL Server 2005) Operations that are not supported by SQL Server replication. For SQL Server 2005 earlier than CU6 for SP2, tables without a declared primary key also are not supported. SQL Server 2005 tables that are in the Extract configuration are marked for SQL Server transactional replication when you enable supplemental logging in Chapter 4. Refer to SQL Server Books Online for a complete list of the operations that are limited by enabling SQL Server Replication.

- (SQL Server 2008) Operations that are not supported by SQL Server Change Data Capture. SQL Server 2008 tables that are in the Extract configuration are marked for SQL Server Change Data Capture when you enable supplemental logging (see page 22). Refer to SQL Server Books Online for a complete list of the operations that are limited by enabling SQL Server Change Data Capture.

- Extraction or replication of DDL (data definition language) operations.

- Extraction from views. The underlying tables can be extracted and replicated.

- Operations by the TextCopy utility and WRITETEXT and UPDATETEXT statements. These features perform operations that either are not logged by the database or are only partially logged, so they cannot be supported by the Extract process.

- Non-native SQL Server transaction log backups, such as those offered by compression utilities. Do not use Oracle GoldenGate Extract on a database where this log backup technology is in use.

- SQL Server 2008 MERGE operations

- SQL Server 2008 compressed log backups

- SQL Server 2008 SPARSE columns

- TDE (Transparent Data Encryption)

- Partitioned tables that have more than one physical layout across partitions

- Partition switching

- Index rebuilds
CHAPTER 2
Installing Oracle GoldenGate

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


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

The installation process takes a short amount of time.

Installation overview

The Oracle GoldenGate that will be configured for SQL Server data capture (Extract configuration) must only be installed on the SQL Server database server. The Oracle GoldenGate that will apply data to a SQL Server target can be installed on any Windows server with the proper client connectivity drivers installed.

To install Oracle GoldenGate, the following steps are required:

Downloading Oracle GoldenGate

Preparing to install Oracle GoldenGate within a cluster

Installing Oracle GoldenGate on Windows

Integrating Oracle GoldenGate into a cluster

Downloading Oracle GoldenGate

Download the appropriate Oracle GoldenGate build to each system that will be part of the Oracle GoldenGate configuration.

2. On the Welcome page:
   - Select your language.
   - Click Continue.
3. On the Export Validation page:
   - Enter your identification information.
   - Accept the Trial License Agreement (even if you have a permanent license).
   - Accept the Export Restrictions.
   - Click Continue.
4. On the Media Pack Search page:
Preparing to install Oracle GoldenGate within a cluster

This topic covers the installation requirements that apply when Oracle GoldenGate will be installed in a cluster environment. Oracle GoldenGate can be used with any cluster-management solution that is Oracle-certified.

Deciding where to install Oracle GoldenGate binaries and files in the cluster

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

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

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

- The Oracle GoldenGate installation must have the same location path on every node
- At minimum, install the following directories on the shared storage to support Oracle GoldenGate recovery requirements. On UNIX or Linux, you can create symbolic links to them from the installation directory on each node.
  - br
  - dirchk
  - dirdat
  - dirtmp
    These directories are among those created when you issue `CREATE SUBDIRS` during installation.
- The parameter files in the `dirprm` directory, if not placed on the shared drive, must be identical on all nodes. To resolve environment settings that must be different from one node to the other, you can set environment settings so they are inherited from the local

NOTE Before installing the software, review the release notes for any new features, new requirements, or bug fixes that affect your current configuration. Review the readme file for known issues.
Manager process or reference a node-specific Oracle GoldenGate macro file. Because this scenario can be difficult to enforce, the inherent concerns can be avoided by storing the parameter files on the shared drive.

See also “Integrating Oracle GoldenGate into a cluster” on page 13 after you install Oracle GoldenGate.

Installing Oracle GoldenGate on Windows

Follow these steps to install Oracle GoldenGate for SQL Server on a Windows system or in the appropriate location in a cluster. See “Preparing to install Oracle GoldenGate within a cluster” on page 10 for more information.

These instructions apply to all versions of SQL Server. Additional database preparation is required before running the Oracle GoldenGate processes. See Chapter 3.

Installing Oracle GoldenGate into a Windows Cluster

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

Installing the Oracle GoldenGate files

1. Unzip the downloaded file(s) by using WinZip or an equivalent compression product.
2. Move the files in binary mode to a folder on the drive where you want to install Oracle GoldenGate. Do not install Oracle GoldenGate into a folder that contains spaces in its name, even if the path is in quotes. For example:
   
   C:\“Oracle_GoldenGate” is not valid.
   
   C:\Oracle_GoldenGate is valid.
3. From the Oracle GoldenGate folder, run the GGSCI program.
4. In GGSCI, issue the following command to create the Oracle GoldenGate working directories.

   CREATE SUBDIRS

5. Issue the following command to exit GGSCI.

   EXIT

Specifying a custom Manager name

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

- You want to use a name for Manager other than the default of GGSMGR.
Installing Oracle GoldenGate

Installing Oracle GoldenGate on Windows

There will be multiple Manager processes running as Windows services on this system. Each Manager on a system must have a unique name. Before proceeding further, note the names of any local Manager services.

To specify a custom Manager name

1. From the directory that contains the Manager program, run GGSCI.
2. Issue the following command.
   
   ```
   EDIT PARAMS ./GLOBALS
   ```
   
   **NOTE** The `./` portion of this command must be used, because the GLOBALS file must reside at the root of the Oracle GoldenGate installation file.

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

   ```
   MGRSERVNAME <name>
   ```

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

Installing Manager as a Windows service

By default, Manager is not installed as a service and can be run by a local or domain account. However, when run this way, Manager will stop when the user logs out. When you install Manager as a service, you can operate it independently of user connections, and you can configure it to start manually or at system start-up.

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

To install Manager as a Windows service

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

   ```
   install <option> [ ... ]
   ```

   Where: `<option>` is one of the following:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADDEVENTS</td>
<td>Adds Oracle GoldenGate events to the Windows Event Manager. By default, Oracle GoldenGate errors are generic. To produce more specific error content, copy the following files from the Oracle GoldenGate installation directory to the <code>SYSTEM32</code> directory.</td>
</tr>
<tr>
<td></td>
<td><code>category.dll</code></td>
</tr>
<tr>
<td></td>
<td><code>ggmsg.dll</code></td>
</tr>
</tbody>
</table>
4. (Windows Server 2008) If Windows User Account Control (UAC) is enabled, you are prompted to allow or deny the program access to the computer. Select Allow to enable the install program to run. This installs the Manager service with a local system account running with administrator privileges. No further UAC prompts will be encountered when running Manager if installed as a service.

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

### Integrating Oracle GoldenGate into a cluster

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

1. Register the Oracle GoldenGate Manager process (and only Manager) as a cluster-managed resource as you would any other application. Manager must be the only Oracle GoldenGate process that the cluster-management software starts and stops, because it is the parent process that manages all other processes.

2. If the cluster uses a virtual IP address, you may need to obtain an available fixed IP address for the Manager process. The VIP must be an available IP address on the public subnet and cannot be determined through DHCP. In the parameter files of the Extract data pumps, specify the VIP of the remote Manager as the input value of the `RMTHOST` parameter. Other Oracle GoldenGate products that access Manager also should use the VIP.

3. Make certain that all nodes in the cluster have synchronized system clocks. The clocks must be synchronized with the clock on the system where Extract is executed. Oracle GoldenGate compares the time of the local system to the commit timestamps to make critical decisions. For information about synchronizing system clocks, consult www.ntp.org or your systems administrator.

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

5. Mount the shared drive on one node only. This prevents processes from being started on another node. Use the same mount point on all nodes.

6. Configure Oracle GoldenGate as directed in this documentation.

Adding Oracle GoldenGate as a Windows cluster resource

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

- In the cluster administrator, add the Manager process to the group that contains the database instance to which Oracle GoldenGate will connect.
- Make sure all nodes on which Oracle GoldenGate will run are selected as possible owners of the resource.
- Make certain the Manager Windows service has the following dependencies (configurable from the Services control panel):
  - The SQL Server resource
  - The disk resource that contains the Oracle GoldenGate directory
  - The disk resource that contains the database transaction log files
  - The disk resource that contains the database transaction log backup files
CHAPTER 3
Preparing the system for Oracle GoldenGate

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.

Choosing Replicat database connection options

The following are the ways in which Replicat can connect to the target database to perform DML operations.

- Connect through ODBC.
- Connect through OLE DB. This is the default and provides slightly better performance than using ODBC.
- Connect through OLE DB as the SQL Server replication user. NOT FOR REPLICATION must be set on IDENTITY columns, foreign key constraints, and triggers.

**NOTE** In all cases, Replicat always uses ODBC to query for metadata.

Review the following guidelines and procedures to evaluate the advantages and disadvantages of these methods before selecting one to use.

Using ODBC or default OLE DB

If Replicat connects through ODBC or the default OLE DB connection, the following limitations apply:

- To keep IDENTITY columns identical on source and target when using ODBC or default OLE DB, Replicat creates special operations in its transaction to ensure that the seeds are incremented on the target. These steps may reduce delivery performance.
- You must adjust or disable triggers and constraints on the target tables to eliminate the potential for redundant operations.

**To use Replicat with ODBC or default OLE DB**

1. To use ODBC exclusively, include the DBOPTIONS parameter with the USEODBC option in the Replicat parameter file. (To use default OLE DB, no parameter is required.)
2. Disable triggers and constraints on the target tables. See “Disabling triggers and cascade constraints on the target” on page 18.
3. To use IDENTITY columns in a bidirectional SQL Server configuration, define the IDENTITY columns to have an increment value equal to the number of servers in the configuration, with a different seed value for each one. For example, a two-server installation would be as follows:
   - Sys1 sets seed value at 1 with an increment of 2.
   - Sys2 sets seed value at 2 with an increment of 2.
A three-server installation would be as follows:

- Sys1 sets seed value at 1 with an increment of 3.
- Sys2 sets seed value at 2 with an increment of 3.
- Sys3 sets seed value at 3 with an increment of 3.


**NOTE** OLE DB uses the ODBC connection settings to derive connection information for OLE DB together with information on which driver to use.

**Using OLE DB with NOT FOR REPLICATION**

If Replicat connects as the SQL Server replication user through OLE DB, and NOT FOR REPLICATION is enabled for IDENTITY, triggers, and foreign key constraints, the following benefits and limitations apply.

- IDENTITY seeds are not incremented when Replicat performs an insert if the IDENTITY property is marked with NOT FOR REPLICATION. You must ensure data integrity by partitioning the IDENTITY values or by configuring the target database as read-only.
- Triggers are disabled for the Replicat user automatically on the target to prevent redundant operations; however triggers fire on the target for other users.
- Foreign key constraints are not enforced on the target for Replicat transactions. CASCADE updates and deletes are not performed. These, too, prevent redundant operations.
- CHECK constraints are not enforced on the target for Replicat transactions. Even though these constraints are enforced on the source before data is captured, consider whether their absence on the target could cause data integrity issues.

**NOTE** Normal IDENTITY, trigger, and constraint functionality remains in effect for any users other than the Replicat replication user.

**To use Replicat with NOT FOR REPLICATION**

1. In SQL Server Management Studio (or other interface) set the NOT FOR REPLICATION flag on the following objects. For active-passive configurations, set it only on the passive database. For active-active configurations, set it on both databases.
   - Foreign key constraints
   - Check constraints
   - IDENTITY columns
   - Triggers (requires textual changes to the definition; see the Microsoft SQL Server documentation for more information.)

2. Partition IDENTITY values or configure the target database as read-only.

3. In the Replicat MAP statements, map the source tables to appropriate targets, and map the child tables that the source tables reference with triggers or foreign-key cascade constraints. Triggered and cascaded child operations are replicated by Oracle GoldenGate, so the referenced tables must be mapped to appropriate targets to preserve data integrity. Include the same parent and child source tables in the Extract TABLE parameters.
Preparing the system for Oracle GoldenGate
Configuring an ODBC connection

NOTE If referenced tables are omitted from the MAP statements, there are no errors to alert you to integrity violations, such as if a row gets inserted into a table that contains a foreign key to a non-replicated table.

4. In the Replicat parameter file, include the DBOPTIONS parameter with the USEREPLICATIONUSER option. For active-passive configurations, use it only on the passive database. For active-active configurations, use it on both databases.

5. Configure an ODBC data source. See “Configuring an ODBC connection”.

Configuring an ODBC connection

Follow these instructions to create a SQL Server system data source name (DSN). A DSN stores information about how to connect to a SQL Server database through ODBC (Open Database Connectivity). Create a DSN on each SQL Server source and target system.

NOTE Replicat will always use ODBC to query the target database for metadata.

To create a SQL Server DSN

1. Run one of the following ODBC clients:
   - If using a 32-bit version of Oracle GoldenGate on a 64-bit system, create the DSN by running the ODBCAD32.EXE client from the %SystemRoot%\SysWOW64 folder.
   - If using a 64-bit version of Oracle GoldenGate on a 64-bit system, create a DSN by running the default ODBCAD32.EXE client in Control Panel>Administrative Tools>Data Sources (ODBC).
   - If using a version of Oracle GoldenGate other than the preceding, use the default ODBC client in Control Panel>Administrative Tools>Data Sources (ODBC).

2. In the ODBC Data Source Administrator dialog box of the ODBC client, select the System DSN tab, and then click Add.

3. Under Create New Data Source, select the correct SQL Server driver as follows:
   - SQL Server 2005: SQL Native Client driver
   - SQL Server 2008: SQL Server Native Client 10.0 driver

4. Click Finish. The Create a New Data Source to SQL Server wizard is displayed.

5. 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 SQL Server instance name.

6. Click Next.

7. For login authentication, select With Integrated Windows authentication for Oracle GoldenGate to use Windows authentication, or select With SQL Server authentication using a login ID and password entered by the user for Oracle GoldenGate to use database credentials. Supply login information if selecting SQL Server authentication.

8. Click Next.
9. 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. Set the other settings to use ANSI.

10. Click **Next**.

11. Leave the next page set to the defaults.

12. Click **Finish**.

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

14. Close the confirmation box and the **Create a New Data Source** box.

15. Repeat this procedure from step 1 on each SQL Server source and target system.

### Preparing tables for processing

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

#### Disabling triggers and cascade constraints on the target

This step applies only if you configured Replicat to connect to the target database through ODBC or the default OLE DB connection. Skip this step if Replicat will be using OLE DB and will operate as the replication user.

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

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

#### Assigning row identifiers

Oracle GoldenGate requires some form of unique row identifier on the source and target tables to locate the correct target rows for replicated updates and deletes. Different SQL Server versions may have different requirements with respect to row identifiers:

- **SQL Server 2005 pre-Cumulative Update 6 for SP2**: All of the source tables must have a *primary key*. This is a requirement of the SQL Server Replication component, which Oracle GoldenGate uses as part of its capture methodology.
Preparing the system for Oracle GoldenGate
Preparing tables for processing

- **SQL Server 2005 with Cumulative Update 6 for SP2 or higher**: Source tables can have any kind of key listed in “How Oracle GoldenGate determines the kind of row identifier to use”.
- **SQL Server 2008**: Source tables can have any kind of key listed in “How Oracle GoldenGate determines the kind of row identifier to use”. If there is no primary key identified on a table and there are fixed-length columns, the length of one of those fixed-length columns must be below 3800 bytes. In addition, limitations that apply to Change Data Capture also apply to the source tables.

### How Oracle GoldenGate determines the kind of row identifier to use

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

1. Primary key
2. First unique key alphanumerically that does not contain a timestamp or non-materialized computed column.
3. If none of the preceding key types exist (even though there might be other types of keys defined on the table) Oracle GoldenGate constructs a pseudo key of all columns that the database allows to be used in a unique key, excluding those that are not supported by Oracle GoldenGate in a key or those that are excluded from the Oracle GoldenGate configuration. For SQL Server, Oracle GoldenGate enforces the length of row data in target tables without a primary key to be less than 8000 bytes.

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

### Using KEYCOLS to specify a custom key

If a table does not have one of the preceding types of row identifiers, or if you prefer those identifiers not to be used, you can define a substitute key if the table has columns that always contain unique values. You define this substitute key by including a KEYCOLS clause within the Extract TABLE parameter and the Replicat MAP parameter. The specified key overrides any existing primary or unique key that Oracle GoldenGate finds. For more information, see the Oracle GoldenGate Windows and UNIX Reference Guide.

### 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.

### Improving IDENTITY replication with array processing

Replicat must continuously toggle IDENTITY_INSERT off and on when applying IDENTITY data to multiple tables in a session, because only one table per session can have IDENTITY_INSERT set to ON. To improve the performance of Replicat in this situation, use the BATCHSQL
parameter. BATCHSQL causes Replicat to use array processing instead of applying SQL statements one at a time.
Preparing for Oracle GoldenGate capture

CHAPTER 4

Preparing for Oracle GoldenGate capture

Setting the database to full recovery model

Oracle GoldenGate requires a SQL Server source database to be set to the full recovery model.

To verify or set the recovery model

1. Connect to the SQL Server instance with SQL Server Management Studio for SQL Server.
2. Expand the Databases folder.
3. Right-click the source database, and then select Properties.
4. Select the Options tab.
5. Under Recovery, set Model to Full if not already.
6. If the database was in Simple recovery or never had a Full database backup, take a Full database backup before starting Extract. See page 25.
7. Click OK.

Backing up the transaction log

The Extract process may occasionally require access to a log backup on the source system. This happens when the required log records are no longer available in the online log and have been moved to log backups.

Oracle GoldenGate requires the log backup files on a source system to meet the following conditions:

- The backup file must remain in the original location where the backup was made. If the log backup files must be removed from this location, you can override this condition by using the TRANLOGOPTIONS parameter with the ALTARCHIVELOGDEST option. This parameter enables Oracle GoldenGate to search for the backup file in a different location, but must only be used when the backup files are not in the original location.
- The backup cannot be striped across multiple files.
- The backup must be made to a DISK device.
- The backup must be a native SQL Server backup made by issuing the BACKUP LOG command (or the corresponding GUI command). Third-party log backup tools are not supported.
- The backup files must be non-encrypted and non-compressed.
- Every log backup must have a distinct file name. Do not overwrite backup files to the same name as old ones.

For optimal performance of the Extract process, do the following:
Retaining the log backups

Retain enough log backups so that if you stop Extract or there is an unplanned outage, Extract can start again from its checkpoints. Extract must have access to the data in the transaction log or a log backup that contains the start of the oldest uncommitted unit of work, and all log backups thereafter.

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

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

To determine where the Extract checkpoints are, use the `INFO EXTRACT` command. For more information, see the Oracle GoldenGate Windows and UNIX Reference Guide.

Enabling supplemental logging

These instructions apply to new installations of Oracle GoldenGate for all supported SQL Server versions. You will enable supplemental logging with the `ADD TRANDATA` command so that Extract can capture the information that is required to reconstruct SQL operations on the target. This is more information than what SQL Server logs by default.

ADD TRANDATA must be issued for all tables that will be replicated with Oracle GoldenGate. It does the following:

- **SQL Server 2005 updated to CU6 for SP2 or later**: ADD TRANDATA calls the `sys.sp_extended_logging` stored procedure.

  **WARNING** For SQL Server with CU6 for SP2 or later, if Extract and transactional replication will co-exist, do not issue ADD TRANDATA for tables that do not have a declared primary key. Oracle GoldenGate enables those tables for replication, but tables without a primary key are not supported by SQL Server transactional replication and will cause the log reader agent to fail.

  - **SQL Server 2005 pre-CU6 for SP2**: ADD TRANDATA creates the following:
    - A replication publication named `<source database name>: GoldenGate<source database name> Publisher`. To view this publication, look under `Replication>Local Publications` in SQL Server Management Studio. This procedure adds the specified table to the publication as an article.
    - A SQL Server Log Reader Agent job for the publication. This job cannot run concurrently with an Extract process in this configuration. To disable this job, see “Managing the secondary truncation point” on page 23.

  - **SQL Server 2008**: ADD TRANDATA enables Change Data Capture (CDC) and creates a minimal Change Data Capture on the specified table.
Oracle GoldenGate does not use the CDC tables other than as necessary to enable supplemental logging.

As part of enabling CDC, SQL Server creates two jobs per database: `<dbname>_capture` and `<dbname>_cleanup`. The `<dbname>_capture` job adjusts the secondary truncation point and captures data from the log to store in the CDC tables. The `<dbname>_cleanup` job ages and deletes data captured by CDC.

Using the `TRANLOGOPTIONS` parameter with the `MANAGESECONDARYTRUNCATIONPOINT` option for Extract removes the `<dbname_capture>` job, preventing the overhead of the job loading the CDC tables.

The alternative (using `TRANLOGOPTIONS` with `NOMANAGESECONDARYTRUNCATIONPOINT`) requires the SQL Server Agent to be running and requires the `<dbname>_capture` and `<dbname>_cleanup` jobs to be retained. You will probably need to adjust the `<dbname>_cleanup` data retention period if the default of three days is not acceptable for storage concerns.

For more information on `TRANLOGOPTIONS`, see “Managing the secondary truncation point” on page 23.

To enable supplemental logging

1. On the source system, run GGSCI.

2. Log into the database from GGSCI.

   \[DBLOGIN SOURCEDB <DSN>[, USERID <user>, PASSWORD <password>]\]

   Where:

   - `<SOURCEDB <DSN>>` is the name of the SQL Server data source.
   - `USERID <user>` is the Extract login and `PASSWORD <password>` is the password that is required if Extract uses SQL Server authentication.

   **NOTE** If using DBLOGIN with a DSN that is using Integrated Windows authentication, the connection to the database for the GGSCI session will be that of the user running GGSCI. In order to issue ADD TRANDATA or DELETE TRANDATA, this user must be a member of the SQL Server sysadmin server role.

3. In GGSCI, issue the following command for each table that is, or will be, in the Extract configuration. You can use a wildcard to specify multiple table names, but not owner names.

   \[ADD TRANDATA <owner>.<table>\]

   **NOTE** The ADD TRANDATA command for SQL Server automatically filters out tables with `dbo.MS` and `dbo.sys` in their names, because it is assumed they are system tables which should not be replicated. If you have tables that contain either of these naming conventions, do not use wildcards to specify them with ADD TRANDATA. Instead, issue the ADD TRANDATA command separately for each one that you want to be replicated.

Managing the secondary truncation point

When you enable supplemental logging with the ADD TRANDATA command for at least one table in a SQL Server database, a secondary truncation point is created in the transaction log.
Preparing for Oracle GoldenGate capture
Managing the secondary truncation point

log that has to be moved for log space to be released as needed, following subsequent log backups. Use the TRANLOGOPTIONS parameter to control whether Extract or SQL Server manages the secondary truncation point. This is a required parameter.

**Oracle GoldenGate manages the secondary truncation point**

Use TRANLOGOPTIONS with the MANAGESECONDARYTRUNCATIONPOINT option if Extract will not be running concurrently (for the same source database) with:

- (SQL Server 2005, any version) SQL Server transactional replication
- (SQL Server 2008) SQL Server transactional replication and/or CDC that is configured for applications other than Oracle GoldenGate

MANAGESECONDARYTRUNCATIONPOINT enables Extract to manage the secondary truncation point. Extract will move the secondary truncation point forward at a defined interval by issuing the following T-SQL statement:

```sql
EXEC sp_repldone @xactid = NULL, @xact_segno = NULL, @numtrans = 0, @time = 0, @reset = 1
```

When Extract manages the secondary truncation point for SQL Server 2005 pre-CU6 for SP2, you must stop and disable the Log Reader Agent job as follows:

1. In SQL Server Management Studio, connect to the SQL Server 2005 instance.
2. Start SQL Server Agent (if not running).
3. Expand the **SQL Server Agent** folder.
4. Expand the **Jobs** folder.
5. Find the job that was created by the ADD TRANDATA command. The name shows the server, the instance, and the database name, plus the number of the publication. It looks similar to the following:

   KTANCO\SQL2005\SQLLBE_SRC-1

6. Right click the job, and then select **Stop Job**.
7. Right click the job again, and then select **Disable**.

   **NOTE** If the Extract process running against a pre-CU6 source is suspended for a longer time than the normal log backup frequency, you will need to re-enable and start the SQL Server Replication Log Reader Agent job temporarily to manage the last distributed transaction. Stop and disable the job before you restart Extract.

For SQL Server 2008, MANAGESECONDARYTRUNCATIONPOINT also routinely checks and deletes the CDC capture job. This prevents change data from being collected for all of the tables in the database, resulting in:

- better performance by Extract
- less storage space used by captured data in the CDC tables
- fewer records in the transaction log

**NOTE** Using TRANLOGOPTIONS MANAGESECONDARYTRUNCATIONPOINT for Extract when either SQL Server transactional replication and/or CDC for
Preparing for Oracle GoldenGate capture
Making a full database backup before you start Oracle GoldenGate

Maintaining the secondary truncation point while Extract is offline

When Extract manages the secondary truncation point and is stopped for a longer period of time than the log backup frequency, data in the transaction log will not be freed after log backups. This causes the transaction log to grow. To allow space to be freed from the log after backups, mark the data as ‘distributed’ by doing one of the following, depending on whether you intend to continue capture with the same Extract or delete that Extract.

To continue capture with the same Extract group

Before starting this Extract group again, retain all of the log backups that contain data that still needs to be processed.

1. To determine the log-read checkpoint that shows the oldest log that Extract needs, use the INFO EXTRACT command with the SHOWCH option in GGSCI.
   
   INFO EXTRACT <group>, SHOWCH

2. Either manually or from within a SQL Server Agent job, move the secondary truncation point by issuing the following T-SQL command against the source database. Run this command approximately every minute until you start Extract.
   
   EXEC sp_repldone @xactid = NULL, @xact_segno = NULL, @numtrans = 0, @time = 0, @reset = 1

3. Stop and disable the SQL Server Agent job or disconnect the query session prior to restarting Extract.

To remove capture with the Extract group

To delete this Extract group, you can disable Oracle GoldenGate capture instead of manually moving the secondary truncation point. To disable Oracle GoldenGate capture, follow the steps for disabling supplemental logging as directed in “Uninstalling Oracle GoldenGate” on page 27.

SQL Server manages the secondary truncation point

Use TRANLOGOPTIONS with the NOMANAGESECONDARYTRUNCATIONPOINT option if Extract will run concurrently (for the same source database) with:

- (SQL Server 2005, any version) SQL Server transactional replication
- (SQL Server 2008) SQL Server transactional replication and/or CDC that is configured for applications other than Oracle GoldenGate

SQL Server will manage the secondary truncation point.

Making a full database backup before you start Oracle GoldenGate

These instructions are for making a full database backup.
Preparing for Oracle GoldenGate capture
Making a full database backup before you start Oracle GoldenGate

To make a full backup of the database

1. Configure Oracle GoldenGate to your requirements, and then return to this procedure when you are ready to begin initial synchronization and start change replication. To configure Oracle GoldenGate, see the Oracle GoldenGate Windows and UNIX Administrator's Guide.

2. Connect to the SQL Server instance with SQL Server Management Studio.

3. Expand the Databases folder.

4. Right click the source database name, and then select All Tasks > Backup Database.

5. Select Database - Complete. This option makes a full database backup and ensures that no transaction information is lost when Oracle GoldenGate starts.

6. Under Destination, click Add to specify the backup file name and location.

7. Click OK. The backup file is added to the Destination list box in the SQL Server Backup dialog box.

8. Click OK to start the backup.
CHAPTER 5
Uninstalling Oracle GoldenGate

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

Disabling supplemental logging

These steps must be performed before you uninstall Oracle GoldenGate.

To disable supplemental logging from SQL Server 2005 pre-CU6 for SP2
1. Stop Extract.
   STOP EXTRACT <group>
2. Run DBCC OPENTRAN against the database to verify that there are no open transactions.
3. Wait for open transactions to commit before proceeding to the next step.
4. Manually delete the Oracle GoldenGate publication from the database.

To disable supplemental logging from SQL Server 2005 with CU6 for SP2 or later

When there are concurrent native SQL Server transactional replication publications for the source database, issue the DELETE TRANDATA command for all of the tables that are listed in the TABLE parameter. A wildcard can be used for the table name, but not the owner name.
The syntax is:
DELETE TRANDATA <owner>.<table>

When there are no concurrent native SQL Server transactional replication publications for the source database, follow these steps:
1. Pause application and transactional activity. There should be no new records occurring between the timing of step 6 and the completion of step 8.
2. Stop Extract.
   STOP EXTRACT <group>
3. Run DBCC OPENTRAN against the database to verify that there are no open transactions.
4. Wait for open transactions to commit before proceeding to the next step.
5. Execute EXECUTE sp_repltrans to verify that there are no undistributed transactions in the database.
Uninstalling Oracle GoldenGate

Removing Oracle GoldenGate from Windows Cluster

6. If the EXECUTE sp_repltrans returned any records, execute the following command against the database to clear any remaining undistributed transactions. Otherwise, skip to step 8.

   EXECUTE sp_repldone @xactid=NULL, @xact_segno=NULL, @numtrans=0, @time=0, @reset=1

7. Execute EXECUTE sp_repltrans again to verify that all transactions are distributed.

8. Issue the DELETE TRANDATA command for all of the tables that are listed in the TABLE parameter in the Extract parameter file. A wildcard can be used for the table name but not the owner name.

   DELETE TRANDATA <owner.table>

To disable supplemental logging from SQL Server 2008

When there are concurrent native SQL Server transactional replication publications or there are Change Data Capture configurations for applications other than Oracle GoldenGate, issue the DELETE TRANDATA command for all of the tables that are listed in the TABLE parameter in the Extract parameter file. A wildcard can be used for the table name but not the owner name. The syntax is:

   DELETE TRANDATA <owner>.<table>

When there are no concurrent native SQL Server transactional replication publications, and there are no Change Data Capture configurations for applications other than Oracle GoldenGate, follow these steps:

1. Verify that there are no open transactions by running DBCC OPENTRAN against the database.
2. Wait for any open transactions to commit before proceeding to the next step.
3. Run the following against the source database:

   EXEC sys.sp_cdc_disable_db

Removing Oracle GoldenGate from Windows Cluster

1. Working from the node in the cluster that owns the cluster group that contains the Manager resource, run GGSCI and then stop any Extract and Replicat processes that are still running.
2. Use the Cluster Administrator tool to take the Manager resource offline.
3. Right click the resource and select Delete to remove it.
4. Click Start > Run, and type cmd in the Run dialog box to open the command console.
5. Change directories to the Oracle GoldenGate installation directory.
6. Run the install program using the following syntax.

   install deleteevents deleteservice

   This command stops Oracle GoldenGate events from being reported to the Windows Event Manager and removes the Manager service.
7. Delete the CATEGORY.DLL and GGMSG.DLL files from the Windows SYSTEM32 folder.
8. Move the cluster group to the next node in the cluster, and repeat from step 4.
9. Follow the instructions in “Uninstalling Oracle GoldenGate from Windows (non-cluster)”.

Uninstalling Oracle GoldenGate from Windows (non-cluster)

1. (Suggested) Log on as the system administrator or as a user with permission to issue Oracle GoldenGate commands and to delete files and directories from the operating system.
2. From the Oracle GoldenGate installation folder, run GGSCI.
3. Stop all Oracle GoldenGate processes.
4. Stop the Manager program or service.
   
   **NOTE** Skip step 5 through step 8 if you already performed them when removing Oracle GoldenGate from a Windows cluster.
5. Click **Start > Run**, and type `cmd` in the **Run** dialog box to open the command console.
6. Change directories to the Oracle GoldenGate installation directory.
7. Run the `install` program using the following syntax.

```
install deleteevents deleteservice
```

This command stops Oracle GoldenGate events from being reported to the Windows Event Manager and removes the Manager service.
8. Delete the CATEGORY.DLL and GGSMSG.DLL files from the Windows SYSTEM32 folder.

**On any system where a Replicat checkpoint table is being used:**
9. Log into the database with the `DBLOGIN` command, and then remove the Replicat checkpoint table by running the `DELETE CHECKPOINTTABLE` command.

**On all systems:**
10. Make certain all processes are stopped (including GGSCI) and then remove the Oracle GoldenGate files by removing the installation directory.
APPENDIX 1

Configuring the replication components (SQL Server 2005 pre-CU6 for SP2)

Follow these instructions only if the following are true:

- Your SQL Server 2005 source is not updated to Microsoft Cumulative Update Package 6 (CU6) for SQL Server 2005 Service Pack 2 or later.
- The SQL Server native replication components and a distribution database are not installed and configured in this SQL Server source.

For a SQL Server 2005 source that is not upgraded to CU6 for SP2, the native replication components must be used with a distribution database to support Oracle GoldenGate replication. Install the SQL Server replication components only if they are not installed and configured already.

What the replication components do for Oracle GoldenGate

- Oracle GoldenGate can operate concurrently with SQL Server 2005 Replication against the same database. Oracle GoldenGate issues a warning message when it detects a Log Reader Agent that is already attached to the database.
- One distribution database can be used for all SQL Server source databases. Oracle GoldenGate does not depend on the distribution database, but instead reads the logs directly, so you can set transaction retention to zero.

Installing and configuring the SQL Server 2005 replication components

These steps install the SQL Server replication components on the local hard drive.

1. On the source system, run Setup.exe in the Servers folder of the SQL Server installation directory.
2. Complete the initial licensing pages.
3. On the Components to Install page, select the database features that you want to install.
4. Click Advanced to open the Feature Selection page.
5. Expand Database Services.
6. Click Replication, and then select Will be installed on local hard drive from the drop-down menu.
7. Click Next.
8. Complete the database setup according to your requirements.
10. Expand the SQL Server instance.
11. Select the Replication folder.
12. Right click Replication, and then select Configure Distribution to start the Configure Distribution wizard.
13. Select the local instance as its own distribution database, or select a remote one.
14. Click Next.
15. Set the SQL Server Agent service to start automatically, if possible.
16. Click Next.
17. Accept the default Snapshot Folder, or choose a new location. Oracle GoldenGate does not use the Snapshot Folder.
18. Click Next.
19. Accept the default database name and file locations, or modify them as needed.
20. Click Finish, and then click Finish again to create the distribution database and finish the setup.

Configuring and cleaning up the distribution database

After the replication components are installed, perform the following steps to:

- Set transaction retention to 0.
- Disable replication alerts.
- Stop and disable SQL Server Agent replication jobs, which are created during the distribution database setup.

To configure transaction retention
1. In SQL Server Management Studio, expand the SQL Server instance.
2. Right-click the Replication folder, and then select Distributor Properties.
3. Click General Properties.
4. To the right of the History Retention column, next to the distribution database name, click the ellipsis (...) button to open the Distribution Database Properties.
5. Set Transaction retention to:
   - At least 0 Hours
   - But not more than 0 Hours
6. On the same page, set History retention to 0.
7. Click OK.

To stop and disable SQL Server Agent replication jobs and alerts
1. In SQL Server Management Studio, connect to the SQL Server instance.
2. Start SQL Server Agent.
3. Expand the SQL Server Agent folder, and then expand the Jobs folder.
4. Right click each of the following jobs, and then select **Stop Job** (if running), then **Disable**.
   - Agent history clean up: <distribution database name>
   - Distribution clean up: <distribution database name>
   - Expired subscription clean up
   - Reinitialize subscriptions having data validation failures
   - Replication agents checkup

5. Under the **SQL Server Agent** folder, expand the **Alerts** folder.

6. Select all alerts that begin with the name “Replication,” and then select Disable.
This appendix describes the programs, directories, and other components created or used by the Oracle GoldenGate software in the Oracle GoldenGate installation directory. Additional files not listed here might be installed on certain platforms. Files listed here might not be installed on every platform.

**Oracle GoldenGate Programs and Utilities**

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

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

**Table 4 Programs and utilities**

<table>
<thead>
<tr>
<th>Program</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cobgen</td>
<td>Generates source definitions based on COBOL layouts. Used for Oracle GoldenGate for Datawise on Stratus.</td>
</tr>
<tr>
<td>convchk</td>
<td>Converts checkpoint files to a newer version.</td>
</tr>
<tr>
<td>ddcob</td>
<td>Generates target DDL table creation statements based on COBOL layouts. Used for Oracle GoldenGate for Datawise on Stratus.</td>
</tr>
<tr>
<td>ddlgen</td>
<td>Generates target database table definitions based on source database DDL. Used primarily on the NonStop platform.</td>
</tr>
<tr>
<td>defgen</td>
<td>Generates data definitions and is referenced by Oracle GoldenGate processes when source and target tables have dissimilar definitions.</td>
</tr>
<tr>
<td>emsclnt</td>
<td>Sends event messages created by Collector and Replicat on Windows or UNIX systems to EMS on NonStop systems.</td>
</tr>
<tr>
<td>extract</td>
<td>Performs capture from database tables or transaction logs or receives transaction data from a vendor access module.</td>
</tr>
<tr>
<td>ggmxinstall</td>
<td>Oracle GoldenGate installation script for the SQL/MX database.</td>
</tr>
<tr>
<td>ggsci</td>
<td>User interface to Oracle GoldenGate for issuing commands and managing parameter files.</td>
</tr>
</tbody>
</table>
Oracle GoldenGate installed components

Oracle GoldenGate subdirectories

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

**NOTE** Some directories may not exist in all installations.

Table 5  Subdirectories

<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>br</td>
<td>Contains the checkpoint files for the bounded recover feature.</td>
</tr>
<tr>
<td>cfg</td>
<td>Contains the property and xml files that are used to configure Oracle GoldenGate Monitor.</td>
</tr>
<tr>
<td>dirdb</td>
<td>Contains the datastore that is used to persist information that is gathered from an Oracle GoldenGate instance for use by the Oracle GoldenGate Monitor application or within Oracle Enterprise Manager.</td>
</tr>
</tbody>
</table>
### Table 5  Subdirectories (continued)

<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
</tr>
</thead>
</table>
| 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. |
Oracle GoldenGate installed components
Oracle GoldenGate subdirectories

<table>
<thead>
<tr>
<th>Directory</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dirpcs</td>
<td>Default location for status files. File name format is &lt;group&gt;.&lt;extension&gt; where &lt;group&gt; is the name of the group and &lt;extension&gt; 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</td>
</tr>
<tr>
<td>dirprm</td>
<td>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 &lt;group name/user-defined name&gt;.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</td>
</tr>
<tr>
<td>dirrec</td>
<td>Not used by Oracle GoldenGate.</td>
</tr>
<tr>
<td>dirrpt</td>
<td>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 &lt;group name&gt;&lt;sequence number&gt;.rpt where &lt;sequence number&gt; is a sequential number appended to aged files. Do not edit these files. Examples: fin2.rpt mgr4.rpt</td>
</tr>
<tr>
<td>dirsql</td>
<td>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.</td>
</tr>
<tr>
<td>dirtmp</td>
<td>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.</td>
</tr>
<tr>
<td>dirwlt</td>
<td>Contains the Oracle Wallet that supports Oracle GoldenGate Monitor. This directory is not installed until the utility that creates the wallet is run.</td>
</tr>
<tr>
<td>UserExitExamples</td>
<td>Contains sample files to help with the creation of user exits.</td>
</tr>
</tbody>
</table>
Other Oracle GoldenGate files

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

NOTE Some files may not be installed in your environment, depending on the database and OS platform.

Table 6  Other files

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bcpfmt.tpl</td>
<td>Template for use with Replicat when creating a run file for the Microsoft BCP/DTS bulk-load utility.</td>
</tr>
<tr>
<td>bcrypt.txt</td>
<td>Blowfish encryption software license agreement.</td>
</tr>
<tr>
<td>cagent.dll</td>
<td>Contains the Windows dynamic link library for the Oracle GoldenGate Monitor C sub-agent.</td>
</tr>
<tr>
<td>category.dll</td>
<td>Windows dynamic link library used by the INSTALL program.</td>
</tr>
<tr>
<td>chkpt_&lt;db&gt;_create.sql</td>
<td>Script that creates a checkpoint table in the local database. A different script is installed for each database type.</td>
</tr>
<tr>
<td>db2cntl.tpl</td>
<td>Template for use with Replicat when creating a control file for the IBM LOADUTIL bulk-load utility.</td>
</tr>
<tr>
<td>ddl_access.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to Microsoft Access DDL.</td>
</tr>
<tr>
<td>ddl_cleartrace.sql</td>
<td>Script that removes the DDL trace file. (Oracle installations)</td>
</tr>
<tr>
<td>ddl_db2.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to DB2 DDL (Linux, UNIX, Windows).</td>
</tr>
<tr>
<td>ddl_db2_os390.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to DB2 DDL (z/OS systems).</td>
</tr>
<tr>
<td>ddl_ddl2file.sql</td>
<td>Script that saves DDL from the marker table to a file.</td>
</tr>
<tr>
<td>ddl_disable.sql</td>
<td>Script that disables the Oracle GoldenGate DDL trigger. (Oracle installations)</td>
</tr>
<tr>
<td>ddl_enable.sql</td>
<td>Script that enables the Oracle GoldenGate DDL trigger. (Oracle installations)</td>
</tr>
<tr>
<td>ddl_filter.sql</td>
<td>Script that supports filtering of DDL by Oracle GoldenGate. This script runs programatically; do not run it manually.</td>
</tr>
<tr>
<td>ddl_informix.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to Informix DDL.</td>
</tr>
<tr>
<td>Component</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>ddl_mss.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to SQL Server DDL.</td>
</tr>
<tr>
<td>ddl_mysql.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to MySQL DDL.</td>
</tr>
<tr>
<td>ddl_nopurgeRecyclebin.sql</td>
<td>Empty script file for use by Oracle GoldenGate support staff.</td>
</tr>
<tr>
<td>ddl_nssql.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.</td>
</tr>
<tr>
<td>ddl_ora9.sql</td>
<td>Scripts that run programmatically as part of Oracle GoldenGate DDL support; do not run these scripts.</td>
</tr>
<tr>
<td>ddl_ora10.sql</td>
<td></td>
</tr>
<tr>
<td>ddl_ora11.sql</td>
<td></td>
</tr>
<tr>
<td>ddl_ora10upCommon.sql</td>
<td></td>
</tr>
<tr>
<td>ddl_oracle.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to Oracle DDL.</td>
</tr>
<tr>
<td>ddl_pin.sql</td>
<td>Script that pins DDL tracing, the DDL package, and the DDL trigger for performance improvements. (Oracle installations)</td>
</tr>
<tr>
<td>ddl_purgeRecyclebin.sql</td>
<td>Script that purges the Oracle recyclebin in support of the DDL replication feature.</td>
</tr>
<tr>
<td>ddl_remove.sql</td>
<td>Script that removes the DDL extraction trigger and package. (Oracle installations)</td>
</tr>
<tr>
<td>ddl_session.sql</td>
<td>Supports the installation of the Oracle DDL objects. This script runs programmatically; do not run it manually.</td>
</tr>
<tr>
<td>ddl_session1.sql</td>
<td></td>
</tr>
<tr>
<td>ddl_setup.sql</td>
<td>Script that installs the Oracle GoldenGate DDL extraction and replication objects. (Oracle installations)</td>
</tr>
<tr>
<td>ddl_sqlmx.tpl</td>
<td>Template used by the DDLGEN utility to convert Tandem Enscribe DDL to NonStop SQL/MX DDL.</td>
</tr>
<tr>
<td>ddl_status.sql</td>
<td>Script that verifies whether or not each object created by the Oracle GoldenGate DDL support feature exists and is functioning properly. (Oracle installations)</td>
</tr>
<tr>
<td>ddl_staymetadata_off.sql</td>
<td>Scripts that control whether the Oracle DDL trigger collects metadata. This script runs programmatically; do not run it manually.</td>
</tr>
<tr>
<td>ddl_staymetadata_on.sql</td>
<td></td>
</tr>
<tr>
<td>ddl_sybase.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to Sybase DDL.</td>
</tr>
</tbody>
</table>
Table 6 Other files (continued)

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ddl_tandem.tpl</td>
<td>Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.</td>
</tr>
<tr>
<td>ddl_trace_off.sql</td>
<td>Scripts that control whether DDL tracing is on or off.</td>
</tr>
<tr>
<td>ddl_trace_on.sql</td>
<td></td>
</tr>
<tr>
<td>ddl_tracelevel.sql</td>
<td>Script that sets the level of tracing for the DDL support feature. (Oracle installations)</td>
</tr>
<tr>
<td>debug files</td>
<td>Debug text files that may be present if tracing was turned on.</td>
</tr>
<tr>
<td>demo_&lt;db&gt;_create.sql</td>
<td>Scripts that create and populate demonstration tables for use with tutorials and basic testing.</td>
</tr>
<tr>
<td>demo_more_&lt;db&gt;_create.sql</td>
<td></td>
</tr>
<tr>
<td>demo_&lt;db&gt;_insert.sql</td>
<td></td>
</tr>
<tr>
<td>demo_more_&lt;db&gt;_insert.sql</td>
<td></td>
</tr>
<tr>
<td>demo_&lt;db&gt;_lob_create.sql</td>
<td></td>
</tr>
<tr>
<td>demo_&lt;db&gt;_misc.sql</td>
<td></td>
</tr>
<tr>
<td>.dmp files</td>
<td>Dump files created by Oracle GoldenGate processes for tracing purposes.</td>
</tr>
<tr>
<td>ENCKEYS</td>
<td>User-created file that stores encryption keys. Written in external ASCII format.</td>
</tr>
<tr>
<td>exitdemo.c</td>
<td>User exit example.</td>
</tr>
<tr>
<td>exitdemo_utf16.c</td>
<td>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.</td>
</tr>
<tr>
<td>FreeBSD.txt</td>
<td>License agreement for FreeBSD.</td>
</tr>
<tr>
<td>ggmmessage.dat</td>
<td>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.</td>
</tr>
<tr>
<td>ggserr.log</td>
<td>File that logs processing events, messages, errors, and warnings generated by Oracle GoldenGate.</td>
</tr>
<tr>
<td>ggsmsg.dll</td>
<td>Windows dynamic link library used by the INSTALL program.</td>
</tr>
<tr>
<td>GLOBALS</td>
<td>User-created file that stores parameters applying to the Oracle GoldenGate instance as a whole.</td>
</tr>
<tr>
<td>help.txt</td>
<td>Help file for the GGSCI command interface.</td>
</tr>
</tbody>
</table>
### Table 6  Other files (continued)

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>icudt38.dll</td>
<td>Windows shared libraries for International Components for Unicode.</td>
</tr>
<tr>
<td>icuin38.dll</td>
<td></td>
</tr>
<tr>
<td>icuuc38.dll</td>
<td></td>
</tr>
<tr>
<td>jagent.bat</td>
<td>Windows batch file for the Java Agent for Oracle GoldenGate Monitor.</td>
</tr>
<tr>
<td>jagent.log</td>
<td>Log files for the Oracle GoldenGate Monitor Agent.</td>
</tr>
<tr>
<td>jagentjni.log</td>
<td></td>
</tr>
<tr>
<td>jagent.sh</td>
<td>UNIX shell script for the Java Agent for Oracle GoldenGate Monitor.</td>
</tr>
<tr>
<td>LGPL.txt</td>
<td>Lesser General Public License statement. Applies to free libraries from the Free Software Foundation.</td>
</tr>
<tr>
<td>libxml2.dll</td>
<td>Windows dynamic link library containing the XML library for the Oracle GoldenGate XML procedures.</td>
</tr>
<tr>
<td>libxml2.txt</td>
<td>License agreement for libxml2.dll.</td>
</tr>
<tr>
<td>marker.hist</td>
<td>File created by Replicat if markers were passed from a NonStop source system.</td>
</tr>
<tr>
<td>marker_remove.sql</td>
<td>Script that removes the DDL marker table. (Oracle installations)</td>
</tr>
<tr>
<td>marker_setup.sql</td>
<td>Script that installs the Oracle GoldenGate DDL marker table. (Oracle installations)</td>
</tr>
<tr>
<td>marker_status.sql</td>
<td>Script that confirms successful installation of the DDL marker table. (Oracle installations)</td>
</tr>
<tr>
<td>notices.txt</td>
<td>Third-party software license file.</td>
</tr>
<tr>
<td>params.sql</td>
<td>Script that contains configurable parameters for DDL support. (Oracle installations)</td>
</tr>
<tr>
<td>pthread-win32.txt</td>
<td>License agreement for pthread-VC.dll.</td>
</tr>
<tr>
<td>pthread-VC.dll</td>
<td>POSIX threads library for Microsoft Windows.</td>
</tr>
<tr>
<td>prvtclkm.plb</td>
<td>Supports the replication of Oracle encrypted data.</td>
</tr>
<tr>
<td>pw_agent_util.bat</td>
<td>Script files that support the Oracle GoldenGate Monitor Agent.</td>
</tr>
<tr>
<td>pw_agent_util.sh</td>
<td></td>
</tr>
<tr>
<td>role_setup.sql</td>
<td>Script that creates the database role necessary for Oracle GoldenGate DDL support. (Oracle installations)</td>
</tr>
</tbody>
</table>
Oracle GoldenGate checkpoint table

When database checkpoints are being used, Oracle GoldenGate creates a checkpoint table with a user-defined name in the database upon execution of the ADD CHECKPOINTTABLE command, or a user can create the table by using the chkpt_<db>_create.sql script, where <db> is the type of database.

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

Table 7  Checkpoint table definitions

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP_NAME (primary key)</td>
<td>The name of a Replicat group using this table for checkpoints. There can be multiple Replicat groups using the same table.</td>
</tr>
<tr>
<td>GROUP_KEY (primary key)</td>
<td>A unique identifier that, together with GROUPNAME, uniquely identifies a checkpoint regardless of how many Replicat groups are writing to the same table.</td>
</tr>
<tr>
<td>SEQNO</td>
<td>The sequence number of the checkpoint file.</td>
</tr>
<tr>
<td>RBA</td>
<td>The relative byte address of the checkpoint in the file.</td>
</tr>
</tbody>
</table>
### Table 7  Checkpoint table definitions

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUDIT_TS</td>
<td>The timestamp of the checkpoint position in the checkpoint file.</td>
</tr>
<tr>
<td>CREATE_TS</td>
<td>The date and time when the checkpoint table was created.</td>
</tr>
<tr>
<td>LAST_UPDATE_TS</td>
<td>The date and time when the checkpoint table was last updated.</td>
</tr>
<tr>
<td>CURRENT_DIR</td>
<td>The current Oracle GoldenGate home directory or folder.</td>
</tr>
</tbody>
</table>
Index

A
ADD TRANDATA command 22, 23
ADDEVENTS Windows service option 12
ADDSERVICE Windows service option 13
AUTOSTART Windows service option 13

B
BATCHSQL parameter 19
before images, logging 22

C
capture job, creating 23
cascade deletes, disabling 18
cascade updates, disabling 18
category.dll 12
Change Data Capture (CDC)
   use by Oracle GoldenGate 22
characters
   multi-byte 6
cleanup job, creating 23
clocks, synchronizing 14
cluster
   installing on 2, 11, 14
   removing from 28
cluster, running in 10
columns
   datatypes supported 6
   number and size supported 7
components, Oracle GoldenGate 33
compressed log backups 8
constraints, integrity
   disabling 18
   requirements 15, 16
CU6 for SP2 update 4

D
data source name (DSN), creating 17
data types, supported 6
database
   distribution, SQL Server 2005 30
   requirements 4
   versions supported 1
DBCC OPENTRAN 27
deleteevents 29
deletes, cascaded 18
deleteservice 29
disk requirements 2
distribution database, configuring 30, 31

E
Enterprise Edition 4

F
files, installed by Oracle GoldenGate 33
firewall, configuring 3
full database backup 4, 25
full recovery model, setting 21

G
ggmessage.dat file 39
GGSMGR default Manager name 11
ggsmsg.dll 12
GLOBALS file 12

I
IDENTITY columns 15, 16, 19
IMAGE data type 6
INFO EXTRACT command 22
installing on
   Linux and UNIX  10
   Windows  11

K
key
   absence of  18, 19
   assigning  18
   TIMESTAMP in  6
KEYCOLUMNS option, TABLE or MAP  19

L
large objects, limitations on  6
libraries, Visual C++  3
LIMITROWS option, DBOPTIONS  19
Linux, installing on  10
log chain, retaining  4
Log Reader Agent job
   creating  22
   disabling  24
logging, supplemental  22
logs
   non-native backups  8
   supplemental  22

M
Manager
   as Windows service  12
   multiple on same system  12
   name, customizing  11
MANAGESECONDARYTRUNCATIONPOINT option  24
MANUALSTART Windows service option  13
Memory requirements for Oracle GoldenGate  1
MERGE operations  8
MGRSERVNAME parameter  12
Microsoft Cumulative Update Package 6 (CU6)  4
Microsoft Visual C++ 2005 SP1 Redistributable Package  3
multi-byte characters  6

N
network configuration  2
NOMANAGESECONDARYTRUNCATIONPOINT option  25
NOT FOR REPLICATION, enabling  16
NTEXT data type  6

O
objects, supported  7
ODBC connection
   about  15
   configuring  17
ODBCAD32.EXE client  17
OLE DB connections, using  15
operating systems supported  1
operations, supported  7
Oracle GoldenGate
   installed programs and files  33
   installing  9
   uninstalling  27

P
Partition switching  8
Partitioned tables  8
PASSWORD Windows service option  13
platforms, supported  1
ports, required by Oracle GoldenGate  3
pre-installation instructions  1
privileges
   database  5
   operating system  3

R
recovery model, setting  21
removing
   Oracle GoldenGate  27
replication components
   configuring  30
   requirements for  4
replication publication, creating  22
replication user, Replicat as  16
replication, disabling  27
rows, size supported  7
Index

S

secondary truncation point, managing 23
spaces
  in folder names 11
sp_repltrans 27
SQL Server Authentication, using 5
SQL Server requirements 4
SQL_Variant data type 6
supplemental logging
  disabling 27
  enabling 22
Supported editions of SQL Server 1
sys.sp_extended_logging stored procedure 22
system requirements 1
system, configuring 15

T

tables
  preparing for processing 18
  supported kinds 7
TCP/IP, configuring 2
TEXT data type 6
TextCopy utility 8
TIMESTAMP data type 6
TRANLOGOPTIONS parameter 23, 24

transaction log
  backing up 21
  backup, retaining 22
transaction retention, configuring 31
Transparent Data Encryption 8
triggers, disabling on target 18
truncation point, managing 23

U

uninstalling Oracle GoldenGate 27
UNIX, installing on 10
UPDATETEXT statement 8
USER Windows service option 13
USERID parameter 5

V

VAMSERV program 34
VARCHAR (MAX) data type 6
virtual machine, support for 3
Visual C ++ 2005 SP1 Redistributable Package 3

W

Windows Authentication, using 5
Windows, installing on 11
WRITETEXT statement 8