Oracle® GoldenGate

Sybase Installation and Setup Guide

Version 10.4

October 2009



Oracle® GoldenGate Sybase Installation and Setup Guide, version 10.4

Copyright © 1995, 2009 Oracle and/or its affiliates. All rights reserved.

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

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

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Chapter 1 System requirements and preinstallation instructions 3

Overview of GoldenGate for Sybase 3
Supported platforms 3
Operating system requirements 3
Database requirements 5
Supported data types 5
Integers 5
Floating-point numbers 6
Character data 6
Dates and timestamps 6
Large objects 7
Money types 7
IDENTITY type 7
Non-supported datatypes 8
Supported operations and objects 8
Non-supported operations and objects 8
Supported and non-supported object names and case 8
Object names and owners 8
Case sensitivity 8
Supported characters 9
Non-supported characters 10

Chapter 2 Installing GoldenGate 11

Installation overview 11
Upgrades 11
New installations 11
Downloading GoldenGate 11
Setting library paths for dynamic builds 12
Installing GoldenGate on Linux and UNIX 13
Installing the GoldenGate files 13
Configuring Manager and other processes 14
Installing GoldenGate on Windows and Windows Cluster 14

Obtaining the Microsoft redistributable package 14
Installing GoldenGate into a Windows Cluster 14
Installing the GoldenGate files 14
Specifying a custom Manager name 15
Installing Manager as a Windows service 15
Adding GoldenGate as a Windows cluster resource 16
Configuring Manager and other processes 20

Chapter 3 Preparing the system for GoldenGate 21

Preparing the transaction logs 21
Enabling transaction logging 21
Sizing and retaining the logs 22
Initializing the secondary truncation point 22
Preparing tables for processing 22
Disabling triggers and cascade delete constraints 23
Assigning row identifiers 23
Constraining updates when a table has no key 24

Chapter 4 Uninstalling GoldenGate 25

Uninstalling GoldenGate from UNIX 25
Uninstalling GoldenGate from Windows (non-cluster) 25
Uninstalling GoldenGate from Windows Cluster 26

Appendix 1 GoldenGate installed components 27

GoldenGate Programs and Utilities 27
GoldenGate subdirectories 29
Other GoldenGate files 31
GoldenGate checkpoint table 35

Index 36

CHAPTER 1

System requirements and preinstallation instructions

Overview of GoldenGate for Sybase

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

Supported platforms

Sybase 12.5.4 and 15 ASE

To find out which GoldenGate builds are available for a specific combination of database version and operating system, go to http://support.goldengate.com. A valid user name and password are required to enter this site.

Operating system requirements

Disk requirements

- Assign the following free disk space:
  - 50 MB for the GoldenGate installation files. 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 GoldenGate that you are installing on the system. For example, to install two builds of GoldenGate into two separate directories, allocate 80 MB of space.
  - Additional disk space on any system that hosts GoldenGate trails, which contain the working data. The space that is consumed by the trails varies, depending on the volume of data that will be processed. A good starting point is 1 GB.
  - To install GoldenGate into a cluster environment, install the GoldenGate binaries and files on a shared file system that is available to all cluster nodes.

TCP/IP

- Configure the system to use TCP/IP services, including DNS.
- Configure the network with the host names or IP addresses of all systems that will be hosting GoldenGate processes and to which GoldenGate will be connecting. Host names are easier to use.
- GoldenGate requires the following unreserved and unrestricted TCP/IP ports:
  - One port for communication between the Manager process and other GoldenGate processes.
  - A range of ports for local GoldenGate communications: can be the default range starting at port 7840 or a customized range of up to 256 other ports.
- Keep a record of the ports you assigned to GoldenGate. You will specify them with parameters when configuring the Manager process.
- Configure your firewalls to accept connections through the GoldenGate ports.
- If possible, grant unrestricted FTP access to GoldenGate for transfers of data, parameters, and reports between source and target systems. Otherwise, provide for another transfer method. A secure transfer method is also required to resolve support cases.
- If possible, provide a connection between your source and target systems and a site where files can be staged for transfer to and from the GoldenGate Software FTP Support Site (ftp://support.goldengate.com).

Operating system permissions

- To install on Windows, the user installing GoldenGate must log in as Administrator.
- To install on UNIX, the user installing GoldenGate must have read and write privileges on the GoldenGate installation directory.
- The GoldenGate processes require an operating system user that has privileges to read, write, and delete files and subdirectories in the GoldenGate directory. In addition, the user for the Manager process requires privileges to control GoldenGate processes.
- The Extract process requires an operating system user that has read access to the transaction log files, both online and archived.
- It is recommended that these operating system users be dedicated to GoldenGate. Sensitive information might be available to anyone running a GoldenGate process.

Virtual machines

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

Database requirements

Database configuration

- Set the DSQUERY variable to the server that contains the database that GoldenGate will be using.
- The Extract process uses an interface that makes calls directly to the Sybase Replication API.
- Extract uses the Sybase LTM to read the Sybase transaction log. Extract cannot run if Sybase RepServer is running. Only one process at a time can reserve a context that allows it to read the transaction log on the same database.
- The Extract process must be permitted to manage the secondary log truncation point.
- The source replication server must be an active database. GoldenGate cannot capture from a database that is in warm standby mode.

Database user

- Create a database user that is dedicated to GoldenGate. It can be the same user for all of the GoldenGate processes that must connect to a database:
  - Extract (source database)
  - Replicat (target database)
  - DEFGEN (source or target database)
- To preserve the security of your data, and to monitor GoldenGate processing accurately, do not permit other users, applications, or processes to log on or operate as the GoldenGate database user.
- The Extract process requires permissions to access the database. You can:
  - Grant System Administrator privileges.

    Or...

  - Assign a user name with the sa_role and replication_role. The command to grant replication role is:

    ```
    sp_role 'grant', replication_role, <Extract user>
    ```

- The Replicat process requires connect and DML privileges on the target database.

Supported data types

Integers

- BIT
- DECIMAL
- INT
- TINYINT
- NUMERIC
- SMALLINT

Limitations of support

- NUMERIC and DECIMAL (fixed-point) are supported with no integrity loss when moving data to a target column of the same data type without involving calculations or transformation. When calculations or transformation must be performed, GoldenGate supports a maximum value of a signed long integer (32-bits).
- BIT is supported for automatic mapping between Sybase databases. To move data between Sybase and another database type, GoldenGate treats BIT data as binary. In this case, the following are required:
  - The BIT column must be mapped to the corresponding source or target column with a COLMAP clause in a TABLE or MAP statement.
  - The BIT data must be converted by means of the NUMBIN GoldenGate column-conversion function.

Floating-point numbers

- DOUBLE
- FLOAT
- REAL

Limitations of support

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. GoldenGate rounds or truncates values that exceed the supported precision.

Character data

- CHAR
- NCHAR
- NVARCHAR
- VARCHAR

Limitations of support

These data types are supported without limitation to the maximum length supported by the database version:

- 255 bytes for ASE 12.0 and earlier
- The maximum page size for ASE 12.5 and later

These size differences become significant when moving data in a bidirectional configuration between a 12.5 version and an earlier one. If corresponding values are larger than 255 bytes in the newer database, there can be integrity issues when those values are replicated to the older version.

Dates and timestamps

- DATETIME
- SMALLDATETIME
- TIME
- DATE

Limitations of support

- The Sybase API that is used by GoldenGate treats a DATE data type as a DATETIME data type. When mapping your data, treat DATE as DATETIME and observe the Sybase range of legal values for DATETIME (January 1, 1753 to December 31, 9999, with a precision of 1/300th of a second (3.33 ms)).

Large objects

- BINARY
- IMAGE
- TEXT
- VARBINARY

Limitations of support

- TEXT and IMAGE are supported up to 2 GB in length.
- Large objects that are replicated from other databases (such as Oracle BLOB and CLOB) can be mapped to Sybase CHAR, VARCHAR, BINARY, and VARBINARY columns. To prevent Replicat from abending if the replicated large object is bigger than the target column size, use the DBOPTIONS parameter with the ALLOWLOBDATATRUNCATE option in the Replicat parameter file. For more information, see the *GoldenGate for Windows and UNIX Reference Guide*.
- To move data to a Sybase target from a source database that permits empty LOB columns, use the DBOPTIONS parameter with the EMPTYLOBSTRING '<string>' option in the Extract parameter file. This parameter accepts a string value and prevents GoldenGate from setting the target column to be NULL, which is not permitted by Sybase. For more information, see the *GoldenGate for Windows and UNIX Reference Guide*.

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

Money types

- MONEY
- SMALLMONEY

Limitations of support

These data types are supported with no integrity loss when moving data to a target column of the same data type, without involving calculations or transformation. When calculations or transformation must be performed, GoldenGate supports a maximum value of a signed long integer (32-bits).

IDENTITY type

The IDENTITY data type is supported for replication in one direction only, but not for a bi-directional configuration.

Non-supported datatypes

- User-defined types
- TIMESTAMP

> **NOTE** Timestamp columns must be excluded from GoldenGate because they are populated automatically by the database, which generates errors on the target if Replicat attempts to apply a replicated timestamp value. To exclude timestamp columns from being captured by GoldenGate, use the COLSEXCEPT option of the TABLE parameter. Because the system generates the timestamps, the source and target values will be different.

Supported operations and objects

- Extraction and replication of insert, update, and delete operations on Sybase tables that contain rows of up to 512 KB in length.
- GoldenGate supports the maximum number of columns per table that is supported by the database. GoldenGate supports the maximum column size that is supported by the database.
- TRUNCATE TABLE is supported if the table names are unique across all schemas. If not, use the IGNORETRUNCATES parameter for those tables to prevent Replicat from abending.

Non-supported operations and objects

- Extraction or replication of DDL (data definition language) operations
- The SQLEXEC feature of GoldenGate is not supported within a TABLE or MAP statement.
- The BATCHSQL feature of GoldenGate is not supported for Sybase.
- Dynamic SQL. When configuring Replicat, include the NODYNSQL parameter in the parameter file.
- Multi-Extract configuration. Only one Extract can reserve a context to read the Sybase transaction logs.

Supported and non-supported object names and case

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

Object names and owners

Source and target object names must be fully qualified in GoldenGate parameter files, as in fin.emp.

Case sensitivity

If a database is case-sensitive, GoldenGate supports the case sensitivity of database names, owner names, object names, column names, and user names.

If a database is case-insensitive, or if it supports case-sensitivity but is configured to be

case-insensitive, GoldenGate converts all names to upper case.

To preserve case-sensitivity

Case-sensitive names must be specified in GoldenGate parameter files exactly as they appear in the database. Enclose case-sensitive names in double quotes if the other database (the source or target of the case-sensitive objects) is not case-sensitive.

If replicating from a case-insensitive database to a case-sensitive database, the source object names must be entered in the Replicat `MAP` statements in upper case, to reflect the fact that they were written to the trail as uppercase by Extract.

For example:

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

Supported characters

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

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

Character	Description
~	Tilde
< >	Greater-than and less-than symbols
/	Forward slash
\	Backward slash
!	Exclamation point
@	At symbol
#	Pound symbol
$	Dollar symbol
%	Percent symbol
^	Carot symbol
( )	Open and close parentheses
_	Underscore
-	Dash
+	Plus sign

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

Character	Description
=	Equal symbol
|	Pipe
[ ]	Begin and end brackets
{ }	Begin and end curly brackets (braces)

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

Non-supported characters

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

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

Character	Description
&	Ampersand
*	Asterisk
?	Question mark
:	Colon
;	Semi-colon
,	Comma
‘ ’	Single quotes
“ ”	Double quotes
‘	Accent mark (Diacritical mark)
.	Period
	Space

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

CHAPTER 2

Installing GoldenGate

Installation overview

These instructions are for installing GoldenGate for the first time. Installing GoldenGate installs all of the components required to run and manage GoldenGate processing (exclusive of any components required from other vendors, such as drivers or libraries) and it installs the GoldenGate utilities. The installation process takes a short amount of time.

Upgrades

To upgrade GoldenGate from one version to another, follow the instructions on the GoldenGate support site at http://download.oracle.com/docs/cd/E15881_01/index.htm.

New installations

To install GoldenGate for the first time, the following steps are required:

- Downloading GoldenGate
- Setting library paths for dynamic builds
- Installing the software

> **NOTE** Before proceeding, make certain that you have reviewed the System Requirements.

Downloading GoldenGate

1. Navigate to http://edelivery.oracle.com.
2. Select a language.
3. Click **Continue**.
4. Fill in the **Export Validation** form and check the box to agree to the license agreement.
5. Click **Continue**.
6. Select the **Oracle Fusion Middleware** Product Pack, and select the platform on which you will be installing the GoldenGate software.
7. Click **Go**. A list of Oracle Fusion Middleware products for the selected platform is displayed.
8. Select the appropriate Oracle GoldenGate Media Pack.

9. Click **Continue**. The Media Pack contents are displayed, including software builds and documentation.
10. Click **Download** next to each desired component, and follow the automatic download process to transfer the mediapack.zip file to your system.

Setting library paths for dynamic builds

As of version 10, GoldenGate uses shared libraries. When installing GoldenGate on a UNIX system, the following must be true *before running GGSCI or any GoldenGate process*.

1. Make certain that the database libraries are added to the system's shared-library environment variables. This procedure is usually performed at database installation time. Consult your Database Administrator if you have any questions.
2. If you will be running a GoldenGate program from outside the GoldenGate installation directory on a UNIX system:
   - (Optional) Add the GoldenGate installation directory to the PATH environment variable.
   - (Required) Add the GoldenGate installation directory to the shared-libraries environment variable.

   For example, given a GoldenGate installation directory of /ggs/10.0, the second command in the following table requires these variables to be set:

Command	Requires GG libraries in environment variable?
$ ggs/10.0 > ./ggsci	No
$ ggs > ./10.0/ggsci	Yes

To set the variables in Korn shell

```
PATH=<installation directory>:$PATH
export PATH
<shared libraries variable>=<absolute path of installation directory>:$<shared libraries variable>
export <shared libraries variable>
```

To set the variables in Bourne shell

```
export PATH=<installation directory>:$PATH
export <shared libraries variable>=<absolute path of installation directory>:$<shared libraries variable>
```

To set the variables in C shell

```
setenv PATH <installation directory>:$PATH
setenv <shared libraries variable> <absolute path of installation directory>:$<shared libraries variable>
```

Where: <shared libraries variable> is one of the following:

UNIX/Linux library path variables per platform

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

[1] A specific platform may or may not be supported by GoldenGate for your database. See the Systems Requirements for supported platforms.

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

NOTE To view the libraries that are required by a GoldenGate process, use the ldd <process> shell command before starting the process. This command also shows an error message for any that are missing.

Installing GoldenGate on Linux and UNIX

Installing the GoldenGate files

1. Extract the Oracle GoldenGate mediapack.zip file on a Windows system by using WinZip or an equivalent compression product. This produces a UNIX .tar file.
2. FTP the .tar file in binary mode to the UNIX system and directory where you want GoldenGate to be installed.
3. Extract the .tar file. The GoldenGate files are extracted into the current directory.

   ```
   tar -xvof <filename>.tar
   ```

4. Run the command shell and change directories to the new GoldenGate directory.
5. From the GoldenGate directory, run the GGSCI program.

   ```
   GGSCI
   ```

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

   ```
   CREATE SUBDIRS
   ```

7. Issue the following command to exit GGSCI.

   ```
   EXIT
   ```

Configuring Manager and other processes

- To use GoldenGate, you must configure the Manager process. You must specify a TCP/IP port for Manager to use, and you can specify optional parameters that control dynamic port assignments, trail file maintenance, and other properties.
- To begin using GoldenGate, you need to create and configure at least one Extract and Replicat group. Your instructions for these groups determine which data to capture and replicate, and how that data is processed.
- To configure these processes, and to customize GoldenGate, see the *GoldenGate for Windows and UNIX Administrator Guide*.

Installing GoldenGate on Windows and Windows Cluster

Obtaining the Microsoft redistributable package

- Before installing 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 the get the right 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.

Installing GoldenGate into a Windows Cluster

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

Installing the 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 GoldenGate. *Do not* install GoldenGate into a folder that contains spaces in its name, for example "GoldenGate Software." GoldenGate relies on path names, but the operating system does not support path names that contain spaces, whether or not they are within quotes.
3. From the GoldenGate folder, run the GGSCI program.

   ```
   GGSCI
   ```

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

   ```
   CREATE SUBDIRS
   ```

5. Issue the following command to exit GGSCI.

   ```
   EXIT
   ```

Specifying a custom Manager name

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

- you want to use a name for Manager other than the default of GGSMGR.
- there will be multiple Manager processes running as Windows services on this system, such as one for the GoldenGate replication software and one for GoldenGate Veridata. Each Manager on a system must have a unique name. Before proceeding further, verify 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
   ```

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 referenced during installation of the Windows service and during data processing.

Installing Manager as a Windows service

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

To install Manager as a Windows service

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

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

Where: <option> is one of the following:

Table 3 INSTALL options

Option	Description
ADDEVENTS	Adds GoldenGate events to the Windows Event Manager. By default, GoldenGate errors are generic. To produce more specific error content, copy the following files from the GoldenGate installation directory to the SYSTEM32 directory. category.dll ggsmsg.dll
ADDSERVICE	Adds Manager as a service by the name specified in the GLOBALS file, if one exists, or by the default of GGSMGR. ADDSERVICE configures the service to run as the Local System account, the standard for most Windows applications because the service can be run independently of user logins and password changes. To run Manager as a specific account, use the USER and PASSWORD options.[1] The service is installed to start at system boot time (see AUTOSTART). To start it after installation, either reboot the system, or start the service manually from the Services applet of the Control Panel.
AUTOSTART	Specifies that the service created with ADDSERVICE is to be started at system boot time. This is the default unless MANUALSTART is used.
MANUALSTART	Specifies that the service created with ADDSERVICE is to be started manually through GGSCI, a script, or the Services applet of the Control Panel. The default is AUTOSTART.
USER <name>	Specifies a domain user account for executing Manager. For <name>, include the domain name, a backward slash, and the user name, for example HEADQT\GGSMGR. By default, the Manager service is installed to use the Local System account.
PASSWORD <password>	Specifies the password for the user specified with USER.

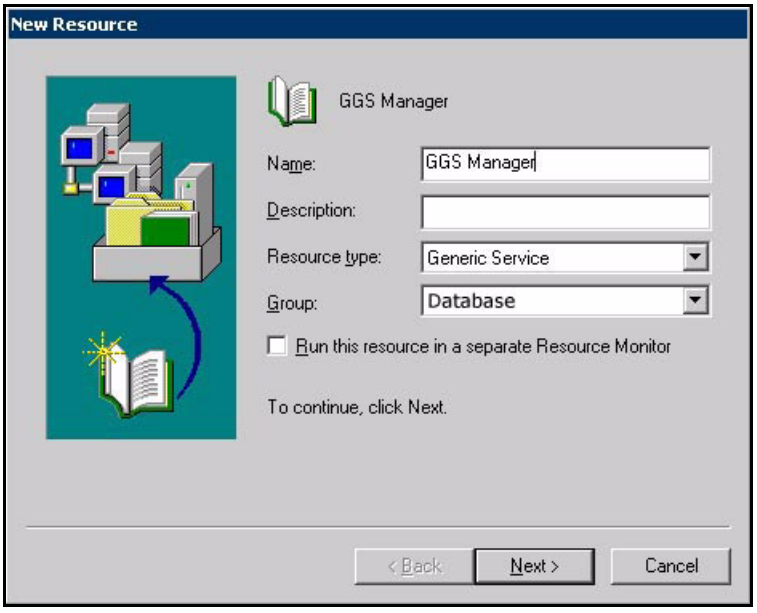
[1] A user account can be changed by selecting the Properties action from the Services applet of the Windows Control Panel.

Adding GoldenGate as a Windows cluster resource

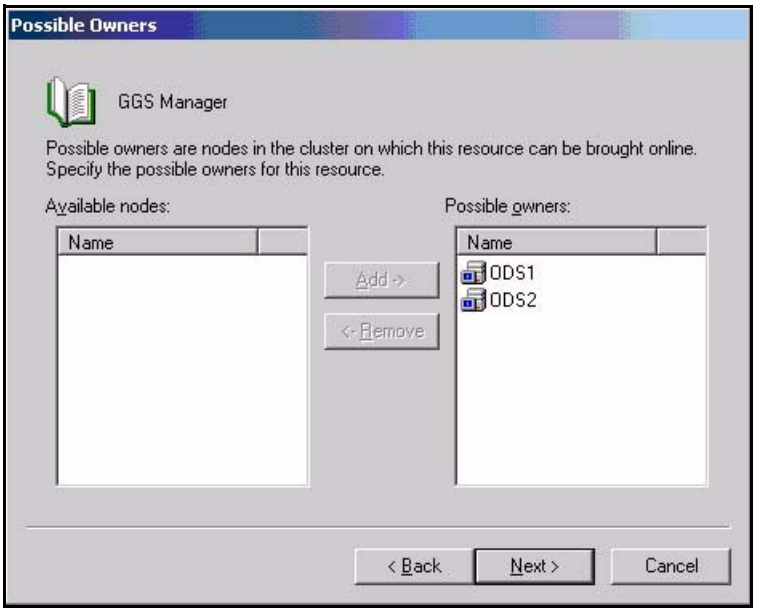
If you installed GoldenGate into a cluster, follow these instructions to establish GoldenGate as a cluster resource and configure the Manager service correctly on all nodes.

1. In the Cluster Administrator, select **File>New>Resource**.

2. In the New Resource dialog box, provide a descriptive name for the GoldenGate Manager (need not be its actual name). For Resource Type, select Generic Service. For Group, select the group that contains the database instance to which GoldenGate will connect.

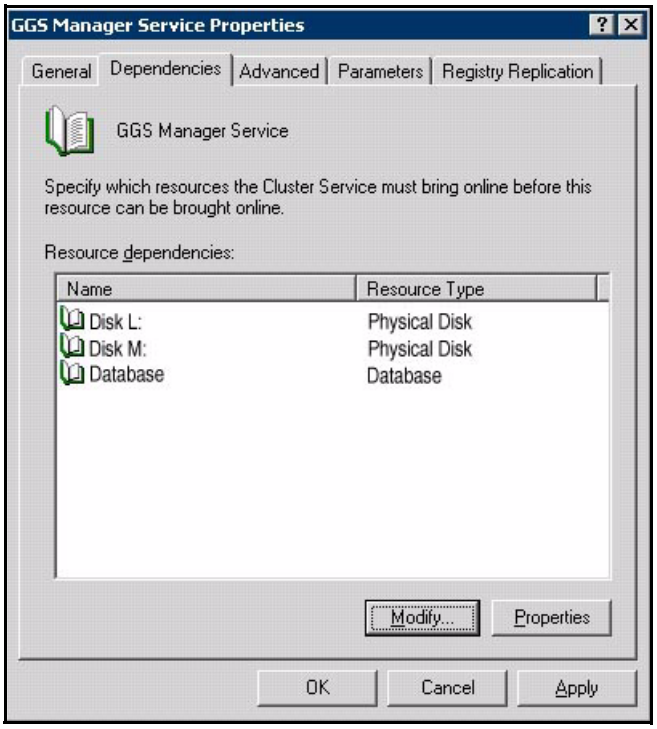


3. Click **Next**.
4. In the Possible Owners dialog box, select the nodes on which GoldenGate will run.

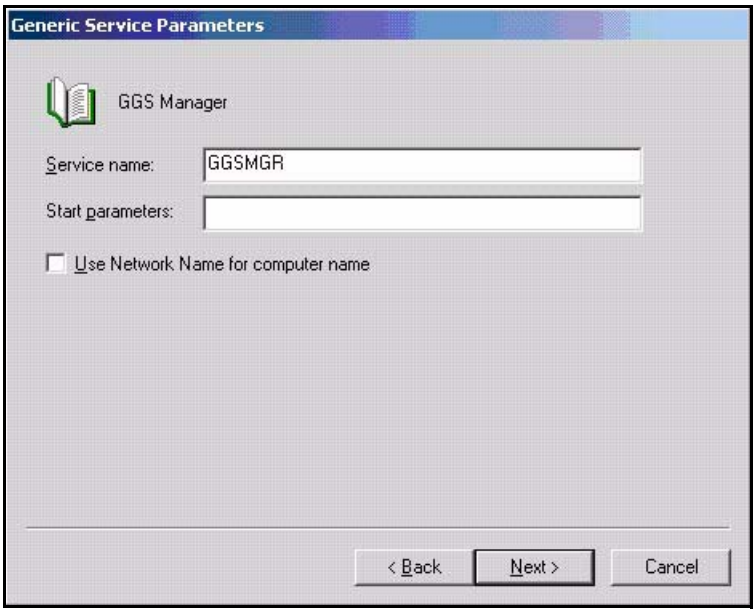


5. Click **Next**.
6. In the GGS Manager Service Properties dialog box, click the Dependencies tab, and add the following to the Resource dependencies list:
   - The database resource group (in this example, it is "Database")

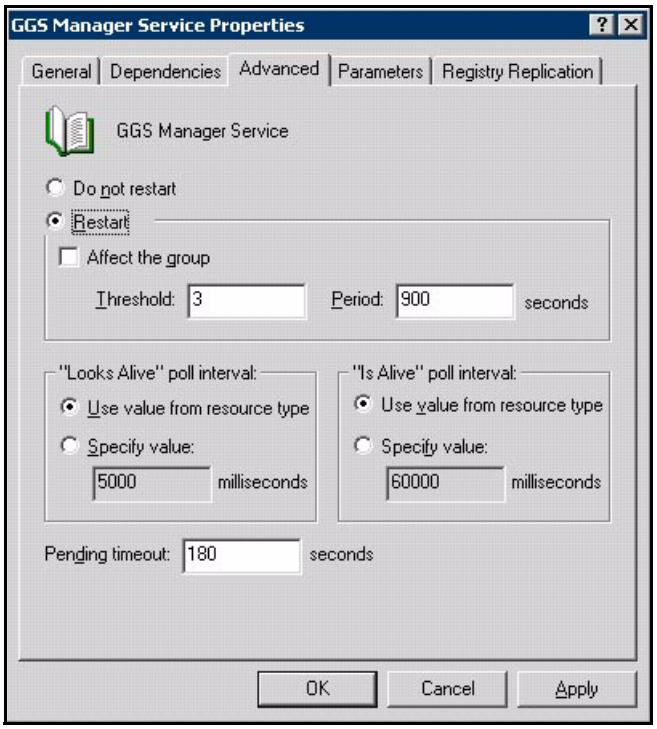
- The disk resource containing the GoldenGate directory
- The disk resource containing the database transaction log files
- The disk resource containing the database transaction log backup files



7. Click **Apply**, then **OK**.
8. In the Generic Service Parameters dialog box, type either the default Manager service name of GGSMGR or, if applicable, the custom name specified in the GLOBALS file.



9. Click **Next**.
10. Click **Finish** to exit the wizard.
11. In the Cluster Administrator tree, right-click the Manager resource and select Properties.
12. Click the Advanced tab, and deselect Affect the Group. This is a recommendation, but you can configure it as needed for your environment.



13. Click **Apply**.
14. Bring the cluster resource online to verify that it was installed correctly.
15. Take the resource offline again.
16. Move the group to the next node in the cluster. When the group has been successfully moved to the second node, the Manager resource should still be offline.
17. Log onto the second node.
18. Install GoldenGate Manager as a service on this node by running the **install** program as you did on the previous node. If you created a custom name for Manager in the GLOBALS file, that name will be used.
19. Bring the resource online to verify that it is running correctly on this node.
20. Repeat steps 18 through 22 for each additional node in the cluster.

Configuring Manager and other processes

- To use GoldenGate, you must configure the Manager process. You must specify a TCP/IP port for Manager to use, and you can specify optional parameters that control dynamic port assignments, trail file maintenance, and other properties.
- To begin using GoldenGate, you need to create and configure at least one Extract and Replicat group. Your instructions for these groups determine which data to capture and replicate, and how that data is processed.
- To configure these processes, and to customize GoldenGate, see the *GoldenGate for Windows and UNIX Administrator Guide*.

CHAPTER 3

Preparing the system for GoldenGate

Preparing the transaction logs

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

Enabling transaction logging

Use the ADD TRANDATA command to mark each source table for replication. This command uses the Sybase sp_setreptable system procedure.

To mark tables for replication

1. On the source system, run GGSCI from the GoldenGate directory.
2. Log into the database from GGSCI.

   ```
   DBLOGIN SOURCEDB <database> USERID <user>
   ```

   Where:

   - <database> is the name of the database.
   - <user> is the database owner or the system administrator. You will be prompted for the password.

3. Issue ADD TRANDATA for each table to be marked.

   ```
   ADD TRANDATA <table> [, LOBSNEVER | LOBSALWAYS | LOBSIFCHANGED]
   ```

   Where:

   - <table> is the name of the table. You can use a wildcard character to specify multiple table names.
   - LOBSNEVER | LOBSALWAYS | LOBSIFCHANGED control whether LOB data is never propagated, only propagated if changes, or always propagated (default).

   NOTE ADD TRANDATA is the recommended way to mark the tables, but the owner or the system administrator can mark them manually by using the Sybase sp_setreptable system procedure. For more information, see the Sybase documentation.

Sizing and retaining the logs

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

If data that Extract needs during processing was not retained, either in online or backup logs, 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 GoldenGate environment over again.

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

Purging log archives

Make certain not to use backup or archive options that cause old archive files to be overwritten by new backups. It is best practice for any new backups to be separate files with different names from older ones. This ensures that if Extract looks for a particular log, it will still exist, and it also ensures that the data is available in case it is needed for a support case.

Initializing the secondary truncation point

Establish a secondary log truncation point prior to running the GoldenGate Extract process. Extract uses the secondary truncation point to identify data that remains to be processed.

To initialize the secondary truncation point

1. Log on to the database as a user with sa_role privileges.
2. Issue the following Sybase command:

```
dbcc settrunc( 'ltm', valid )
```

By default, Extract will manage the secondary truncation point once it is established. Do not permit Extract to be stopped any longer than necessary; otherwise the log could eventually fill up and the database will halt. The only way to resolve this problem is to disable the secondary truncation point and purge the transaction log data. Data not yet processed by Extract will be lost, and you will have to resynchronize the source and target data.

To have Sybase manage the secondary truncation point, use the TRANLOGOPTIONS parameter with the NOMANAGESECONDARYTRUNCATIONPOINT parameter.

Preparing tables for processing

The following table attributes must be addressed in a GoldenGate environment.

Disabling triggers and cascade delete constraints

Disable triggers and cascade delete constraints on target tables, or alter them to ignore changes made by the GoldenGate database user. GoldenGate replicates DML that results from a trigger or cascade delete 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. 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.

You can configure Replicat to disable target triggers at the start of its database session.

To disable triggers with Replicat

1. Assign the Replicat user the replication role.
2. Add the following parameter statement to the root level of the Replicat parameter file.

```
SQLEXEC "set triggers off"
```

Assigning row identifiers

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

How GoldenGate determines the kind of row identifier to use

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) 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 GoldenGate in a key or those that are excluded from the GoldenGate configuration.

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

How to specify your own key for GoldenGate to use

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

Constraining updates when a table has no key

If a target table has no primary key or unique key, duplicate rows can exist. It is possible for GoldenGate to update or delete too many rows in the target 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 GoldenGate on the target system because only one row is processed.

CHAPTER 4

Uninstalling GoldenGate

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

Uninstalling GoldenGate from UNIX

1. Run the command shell.
2. (Suggested) Log on as the system administrator, or as a user with permission to issue GoldenGate commands, and to delete files and directories from the operating system.
3. Change directories to the GoldenGate installation directory.
4. Run GGSCI.
5. Stop all GoldenGate processes.
6. Stop the Manager process.
7. Exit GGSCI.
8. Remove the GoldenGate files by removing the installation directory.
9. Drop any GoldenGate-related objects from the database as needed.

Uninstalling GoldenGate from Windows (non-cluster)

1. (Suggested) Log on as the system administrator, or as a user with permission to issue GoldenGate commands, and to delete files and directories from the operating system.
2. From the GoldenGate installation folder, run GGSCI.
3. Stop all GoldenGate processes.
4. Stop the Manager program or service.
5. Exit GGSCI.
6. Click **Start** > **Run**, and type cmd in the **Run** dialog box.
7. Change directories to the GoldenGate installation directory.

8. Run the install program using the following syntax.

   ```
   install deleteevents deleteservice
   ```

   This command deletes GoldenGate events from being reported to the Windows Event Manager and removes the GoldenGate Manager service.
9. Delete the CATEGORY.DLL and GGSMSG.DLL files from the Windows SYSTEM32 folder.
10. Delete the GoldenGate installation folder.
11. Drop any GoldenGate-related objects from the database as needed.

Uninstalling GoldenGate from Windows Cluster

1. Working from the node in the cluster that owns the cluster group containing 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. Run the install program using the following syntax.

   ```
   install deleteevents deleteservice
   ```

   This command deletes GoldenGate events from being reported to the Windows Event Manager and removes the GoldenGate Manager service.
5. Delete the CATEGORY.DLL and GGSMSG.DLL files from the Windows SYSTEM32 folder.
6. Move the cluster group to the next node in the cluster, and repeat from step 4.
7. Delete the GoldenGate installation folder.
8. Drop any GoldenGate-related objects from the database as needed.

APPENDIX 1

GoldenGate installed components

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

GoldenGate Programs and Utilities

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

Table 4 Programs and utilities

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

Table 4 Programs and utilities (continued)

Program	Description
ggsmgr.jcl ggsmgr.proc ggsmgrst.jcl ggsmgrst.proc	Start the GoldenGate Manager process from a batch job or the operator console on a z/OS system.
install	Installs GoldenGate as a Windows service and provides other Windows-based service options.
keygen	Generates data-encryption keys.
logdump	A utility for viewing and saving information stored in extract trails or files.
mgr	(Manager) Control process for resource management, control and monitoring of GoldenGate processes, reporting, and routing of requests through the GGSCI interface.
replicat	Applies data to target database tables.
reverse	A utility that reverses the order of transactional operations, so that Replicat can be used to back out changes from target tables, restoring them to a previous state.
server	The Collector process, an Extract TCP/IP server collector that writes data to remote trails.
triggen	Generates scripts that create the GoldenGate log table and logging triggers to support the trigger-based extraction method.
vamserv	Started by Extract to read the TMF audit trails generated by TMF-enabled applications using the NonStop SQL/MX database.

GoldenGate subdirectories

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

Table 5 Subdirectories

Directory	Description
dirchk	Contains the checkpoint files created by Extract and Replicat processes, which store current read and write positions to support data accuracy and fault tolerance. Written in internal 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 GoldenGate trail files and extract files created by Extract processes to store records of extracted data for further processing, either by the Replicat process or another application or utility. Written in internal 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 GoldenGate technical support. Example: defs.dat
dirout	This directory is not used any more.

Table 5 Subdirectories (continued)

Directory	Description
dirpcs	Default location for status files. File name format is <group>.<extension> where <group> is the name of the group and <extension> is either pce (Extract), pcr (Replicat), or pcm (Manager). These files are only created while a process is running. The file shows the program name, the process name, the port number, and the process ID. Do not edit these files. Examples: mgr.pcm ext.pce
dirprm	The default location for GoldenGate parameter files created by GoldenGate users to store run-time parameters for GoldenGate process groups or utilities. Written in external ASCII format. File name format is <group name/user-defined name>.prm or mgr.prm. These files may be edited to change GoldenGate parameter values. They can be edited directly from a text editor or by using the EDIT PARAMS command in GGSCI. Examples: defgen.prm finance.prm
dirrec	Not used by GoldenGate.
dirrpt	The default location for process report files created by Extract, Replicat, and Manager processes to report statistical information relating to a processing run. Written in external ASCII format. File name format is <group name><sequence number>.rpt where <sequence number> is a sequential number appended to aged files. Do not edit these files. Examples: fin2.rpt mgr4.rpt
dirsql	The default location for scripts created by the TRIGGEN utility to contain SQL syntax for creating GoldenGate logging triggers and GoldenGate log tables. Written in external ASCII format. File name format is a user-defined name or the defaults of GGSLOG (table-creation script) or the table name (trigger-creation script), with the extension of .sql. These scripts can be edited if needed. Examples: ggslog.sql account.sql

Table 5 Subdirectories (continued)

Directory	Description
dirtmp	The default location for storing large transactions when the size exceeds the allocated memory size. Do not edit these files.
dirver	A GoldenGate Veridata directory. Not used unless this software is installed in the GoldenGate location.

Other GoldenGate files

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

Table 6 Other files

Component	Description
bcpfmt.tpl	Template for use with Replicat when creating a run file for the Microsoft BCP/DTS bulk-load utility.
blowfish.txt	Blowfish encryption software license agreement.
category.dll	Windows dynamic link library used by the INSTALL program.
chkpt_<db>_create.sql	Script that creates a checkpoint table in the local database. A different script is installed for each database type.
db2cntl.tpl	Template for use with Replicat when creating a control file for the IBM LOADUTIL bulk-load utility.
ddl_access.tpl	Template used by the DDLGEN utility to convert source DDL to Microsoft Access DDL.
ddl_cleartrace.sql	Script that removes the DDL trace file. (Oracle installations)
ddl_db2.tpl	Template used by the DDLGEN utility to convert source DDL to DB2 DDL (Linux, UNIX, Windows).
ddl_db2_os390.tpl	Template used by the DDLGEN utility to convert source DDL to DB2 DDL (z/OS systems).
ddl_disable.sql	Script that disables the GoldenGate DDL trigger. (Oracle installations)
ddl_enable.sql	Script that enables the GoldenGate DDL trigger. (Oracle installations)

Table 6 Other files (continued)

Component	Description
ddl_informix.tpl	Template used by the DDLGEN utility to convert source DDL to Informix DDL.
ddl_mss.tpl	Template used by the DDLGEN utility to convert source DDL to SQL Server DDL.
ddl_mysql.tpl	Template used by the DDLGEN utility to convert source DDL to MySQL DDL.
ddl_nssql.tpl	Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.
ddl_ora9.sql	A script that gets tablespace information from an Oracle 9 database.
ddl_ora10.sql	A script that disables the Oracle recyclebin and gets tablespace information from an Oracle 10 database.
ddl_oracle.tpl	Template used by the DDLGEN utility to convert source DDL to Oracle DDL.
ddl_pin.sql	Script that pins DDL tracing, the DDL package, and the DDL trigger for performance improvements. (Oracle installations)
ddl_remove.sql	Script that removes the DDL extraction trigger and package. (Oracle installations)
ddl_setup.sql	Script that installs the GoldenGate DDL extraction and replication objects. (Oracle installations)
ddl_sqlmx.tpl	Template used by the DDLGEN utility to convert Tandem Enscribe DDL to NonStop SQL/MX DDL.
ddl_status.sql	Script that verifies whether or not each object created by the GoldenGate DDL support feature exists and is functioning properly. (Oracle installations)
ddl_sybase.tpl	Template used by the DDLGEN utility to convert source DDL to Sybase DDL.
ddl_tandem.tpl	Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.
ddl_tracelevel.sql	Script that sets the level of tracing for the DDL support feature. (Oracle installations)
debug files	Debug text files that may be present if tracing was turned on.

Table 6 Other files (continued)

Component	Description
demo_<db>_create.sql	Script that creates demonstration tables in the database associated with the GoldenGate installation.
demo_<db>_insert.sql	Script that inserts initial test data into the demonstration tables.
demo_<db>_misc.sql	Script that simulates transaction activity on the demonstration tables.
ENCKEYS	User-created file that stores encryption keys. Written in external ASCII format.
exitdemo.c	User exit example.
ggmessage.dat	Data file that contains error, informational, and warning messages that are returned by the GoldenGate processes. The version of this file is checked upon process startup and must be identical to that of the process in order for the process to operate.
ggserr.log	File that logs processing events, messages, errors, and warnings generated by GoldenGate.
ggsmsg.dll	Windows dynamic link library used by the INSTALL program.
GLOBALS	User-created file that stores parameters applying to the GoldenGate instance as a whole.
help.txt	Help file for the GGSCI command interface.
LGPL.txt	Lesser General Public License statement. Applies to free libraries from the Free Software Foundation.
libodbc.so	ODBC file for Ingres 2.6 on Unix.
libodbc.txt	License agreement for libodbc.so.
libxml2.dll	Windows dynamic link library containing the XML library for GoldenGate's XML procedures.
libxml2.txt	License agreement for libxml2.dll.
marker.hist	File created by Replicat if markers were passed from a NonStop source system.
marker_remove.sql	Script that removes the DDL marker table. (Oracle installations)
marker_setup.sql	Script that installs the GoldenGate DDL marker table. (Oracle installations)

Table 6 Other files (continued)

Component	Description
marker_status.sql	Script that confirms successful installation of the DDL marker table. (Oracle installations)
odbcinst.ini	Ingres 2.6 on Unix ODBC configuration file.
params.sql	Script that contains configurable parameters for DDL support. (Oracle installations)
pthread-win32.txt	License agreement for pthread-VC.dll.
pthread-VC.dll	POSIX threads library for Microsoft Windows.
role_setup.sql	Script that creates the database role necessary for GoldenGate DDL support. (Oracle installations)
sampleodbc.ini	Sample ODBC file for Ingres 2.6 on UNIX.
sqlldr.tpl	Template for use with Replicat when creating a control file for the Oracle SQL*Loader bulk-load utility.
start.prm stop.prm	z/OS paramlib members to start and stop the Manager process.
startmgr stopmgr	z/OS Unix System Services scripts to start the Manager process from GGSCI.
startmgrcom stopmgrcom	z/OS system input command for the Manager process.
tcperrs	File containing user-defined instructions for responding to TCP/IP errors.
usrdecs.h	Include file for user exit API.
zlib.txt	License agreement for zlib compression library.

GoldenGate checkpoint table

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

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

Table 7 Checkpoint table definitions

Column	Description
GROUP_NAME (primary key)	The name of a Replicat group using this table for checkpoints. There can be multiple Replicat groups using the same table.
GROUP_KEY (primary key)	A unique identifier that, together with GROUPNAME, uniquely identifies a checkpoint regardless of how many Replicat groups are writing to the same table.
SEQNO	The sequence number of the checkpoint file.
RBA	The relative byte address of the checkpoint in the file.
AUDIT_TS	The timestamp of the checkpoint position in the checkpoint file.
CREATE_TS	The date and time when the checkpoint table was created.
LAST_UPDATE_TS	The date and time when the checkpoint table was last updated.
CURRENT_DIR	The current GoldenGate home directory or folder.

Index

Symbols

$LD_LIBRARY_PATH variable 13
$PATH variable 12

A

ADD TRANDATA command 21
ADDEVENTS Windows service option 16
ADDSERVICE Windows service option 16
ALLOWLOBDATATRUNCATE option, DBOPTIONS 7
AUTOSTART Windows service option 16

B

BATCHSQL parameter 8
BINARY data type 7
BIT data type 5

C

cascade deletes, disabling 23
case, supported 8
category.dll 16
CHAR data type 6
characters
- supported in object names 9
- wildcard in ADD TRANDATA 21

cluster, installing on 3, 14, 16
columns
- number and size supported 8
- supported data types 5

components, GoldenGate 27
connections, to database 5
constraints, integrity 23
CREATE SUBDIRS command 13

D

data types, supported 5
database requirements 5
DATE data type 6
DATETIME data type 6
DBOPTIONS parameter 7
DDL 8
DECIMAL data type 5
deletes, cascaded 23
disk requirements 3
DOUBLE data type 6
downloading GoldenGate 11
DSQUERY variable 5
dynamic SQL 8

E

EMPTYLOBSTRING option, DBOPTIONS 7
environment variables, setting 12
Extract processes, number of 8

F

files, installed by GoldenGate 27
firewall, configuring 4
FLOAT data type 6
floating-point numbers 6
FTP access for GoldenGate 4

G

ggmessage.dat file 33
GGSMGR default Manager name 15
ggsmsg.dll 16
GLOBALS file 15

GoldenGate
downloading 11
installed programs and files 27
installing 11
uninstalling 25

I

IDENTITY data type 7
IGNORETRUNCATES parameter 8
IMAGE data type 7
INFO EXTRACT command 22
installing GoldenGate 11
installing on
Linux and UNIX 13
Windows 14
INT data type 5

K

key
assigning 23
name, supported characters 9
tables without 24
KEYCOLS option, TABLE or MAP 24

L

large objects, limitations on 7
LIBPATH variable 13
libraries, Visual C++ 14
LIMITROWS option, DBOPTIONS 24
Linux, installing on 13
LOBSNEVER, ALWAYS, IFCHANGED options, ADD TRANDATA 21
logging, enabling 21
logs, preparing 21
LTM, use of 5

M

Manager
as Windows service 15
multiple on same system 15
name, customizing 15
MANUALSTART Windows service option 16
MGRSERVNAME parameter 15
Microsoft Visual C ++ 2005 SP1 Redistributable Package 14
MONEY data type 7

N

name
non-supported characters in 10
supported characters in 9
names, supported 8
NCHAR data type 6
NODYNSQL parameter 8
NUMERIC data type 5
NVARCHAR data type 6

O

objects supported 8
operations, supported 8

P

PASSWORD Windows service option 16
permissions, required 5
ports, required by GoldenGate 4

R

REAL data type 6
removing
constraints from target 22
GoldenGate from system 25
logs 22
replication, marking tables for 21
RepServer, and GoldenGate 5

S

secondary truncation point 5, 22
SHLIB_PATH variable 13
SMALLDATETIME data type 6
SMALLINT data type 5
SMALLMONEY data type 7
sp_setreptable procedure 21

spaces
in folder names 14
SQLEXEC 8
subdirectories, creating 13

T

tables
marking for replication 21
preparing for processing 22
supported size 8
TCP/IP, configuring 4
TEXT data type 7
TIME data type 6
TIMESTAMP data type 8
TINYINT data type 5
transaction logs, preparing 21
triggers, disabling on target 23
TRUNCATE TABLE 8
truncation point, secondary 22

U

uninstalling GoldenGate 25
UNIX, installing on 13
USER Windows service option 16
user, GoldenGate 5
User-defined types 8
user-defined types 6

V

VAMSERV program 28
VARBINARY data type 7
VARCHAR data type 6
virtual machine, support for 4
Visual C ++ 2005 SP1 Redistributable Package 14

W

Windows, installing on 14