Oracle® GoldenGate Veridata

Administrator’s Guide

version 3.0

E17810-02

December 2010

Oracle GoldenGate Veridata Administrator's Guide, version 3.0

Contents

PREFACE

New in Oracle GoldenGate Veridata version 3.0

Expanded database support

- Oracle GoldenGate Veridata Server now supports Microsoft SQL Server as a repository database.
- Oracle GoldenGate Veridata Agent now supports the following databases
  - Microsoft SQL Server
  - Teradata
- Heterogeneous comparisons now can be performed between the different kinds of databases that are supported by Oracle GoldenGate Veridata Agent. The concept of the *comparison format* has been introduced to support heterogeneous data type mapping where source and target data types are different, but similar. During system-generated (automatic) mapping, the comparison format is used, rather than the data type name, to determine whether or not the source and target columns are compatible for comparison. The server process selects the correct format from a set of them that is supported for a given data type and, when applicable, determines the correct precision and scale. A user can override the system selection to specify another supported comparison format. For more information about this feature, see the online help.
- The agent component is now provided in Java code. There is one agent executable for all supported databases. The exception is the NonStop platform: the C-agent is still required for Enscribe and NonStop SQL/MP, and also can be used for Oracle like-to-like comparisons.

Oracle enhancements

- The installation now includes the Oracle client software for use with an Oracle repository.
- Oracle EZCONNECT is now supported for repository connections. The installer prompts for either EZCONNECT or TNSNAMES.

NSK enhancements

- Delta processing now can be enabled or disabled from the Existing Compare Pairs tab of the Compare Pair Configuration page, instead of having to go to the Manual Mapping tab and then select from the drop-down list.

Other enhancements

- The Java agent does not require a GoldenGate Manager process. There is an installer wizard for Windows systems that collects all of the required installation information, including the Windows service. For UNIX systems, the agent is installed in a simple extraction process, with a properties file that requires minor editing to specify some parameter values. (This properties file also underlies the Windows installation.)
- Oracle GoldenGate Veridata Web has a new skin that makes it easier for the eye to move around the sections of a page. Included in the improvements are enhanced status icons and a refined navigation bar that does not require the nodes to be expanded.
- Firefox 3.0 is now supported.
- Oracle GoldenGate Veridata now supports case-sensitive name mapping while doing pattern mapping of compare pairs. First a case-sensitive match is tried, then a case-insensitive match is tried for any objects that still remain unpaired.
- The out-of-sync report and the Out-of-Sync Rows view in the web interface now display the values of all of the out-of-sync columns that are in a table or file.
- To support the addition of a comparison format mapping function (for specifying a standard datatype to which native datatypes are mapped), the Edit Connection page now has three tabs: Connection Details, where the connection information is stored; Connection Settings, where connection parameters reside; and Connection Properties, where the comparison formats are specified.
- From the Details for Finished Compare Pair page, you can now go directly to the Compare Pair Configuration page for that pair.
- Some columns and sections on some pages have been relocated for better workflow, without changing functionality.

CHAPTER 1

About Oracle GoldenGate Veridata

Oracle GoldenGate Veridata compares one set of data to another and identifies data that is out of synchronization. Oracle GoldenGate Veridata supports high-volume, 24x7 replication environments where downtime to compare data sets is not an option. By accounting for data that is being replicated while a comparison takes place, Oracle GoldenGate Veridata can run concurrently with data transactions and replication, while still producing an accurate comparison report.

Oracle GoldenGate Veridata will map column data types across different types of databases automatically, or you can map columns manually in cases where the automatic mapping is not sufficient to accommodate format differences in a heterogeneous environment. For detailed information about this feature, see the online help.

For the purposes of this documentation, the following terms are considered synonymous:

- tables and files
- columns and fields
- rows and records

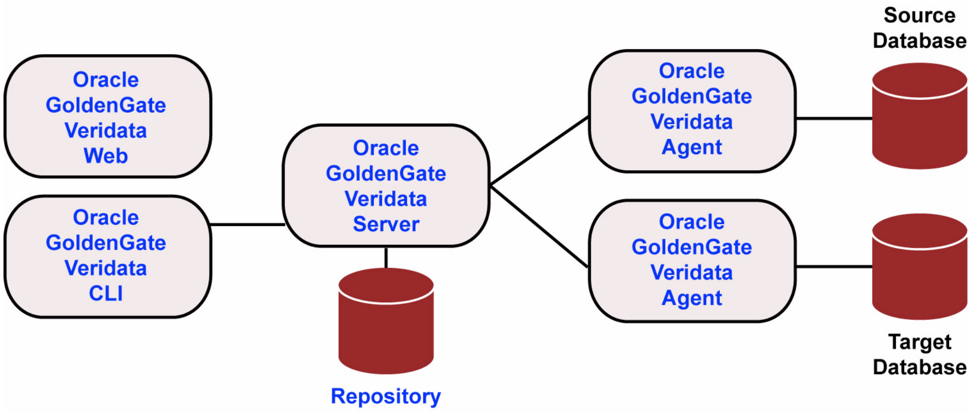
Oracle GoldenGate Veridata Web
Oracle GoldenGate Veridata CLI
Oracle GoldenGate Veridata Server
Repository
Oracle GoldenGate Veridata Agent
Oracle GoldenGate Veridata Agent
Source Database
Target Database

Figure 1 Oracle GoldenGate Veridata components

Oracle GoldenGate Veridata Server

The Oracle GoldenGate Veridata Server performs the following functions:

- Coordinate the execution of Oracle GoldenGate Veridata tasks
- Sort rows (optional)
- Compare data
- Confirm out-of-sync data
- Produce a report for review

Oracle GoldenGate Veridata Web

Oracle GoldenGate Veridata Web is a client to Oracle GoldenGate Veridata Server. It is a browser-based graphical user interface for these activities:

- Configure comparison objects and rules
- Initiate comparisons
- Review the status and output of comparisons
- Review out-of-sync data

Oracle GoldenGate Veridata Repository

The Oracle GoldenGate Veridata repository is a collection of database objects that persists configuration information to disk, saving it permanently as a user environment.

Oracle GoldenGate Veridata Agent

The Oracle GoldenGate Veridata Agent executes the following database-related requests on behalf of the Oracle GoldenGate Veridata Server:

- Fetch and return blocks of rows to compare
- Return column-level detail for out-of-sync rows

Oracle GoldenGate Veridata Manager

The Manager process is part of the C-code based agent that is required for the NonStop platform and optional for the Oracle database. It controls the Oracle GoldenGate Veridata Agent process. A Manager is not used in a Java agent, which is used for the other databases that are supported by Oracle GoldenGate Veridata.

Oracle GoldenGate Veridata CLI (Command Line Interface)

Oracle GoldenGate Veridata includes a command line client to Oracle GoldenGate Veridata Server. The program name is vericom. This interface enables you to run comparisons, specify overrides, and stop Oracle GoldenGate Veridata Server by using automated programs. Configuration and monitoring functions must be performed through Oracle GoldenGate Veridata Web.

Oracle GoldenGate Veridata comparison objects

To begin using Oracle GoldenGate Veridata, you need to create some objects that identify the data that you want to compare and help you to manage your work.

- Configure datasource connections: Oracle GoldenGate Veridata Server connects to Oracle GoldenGate Veridata Agents that interact with the databases that contain the data that is to be compared. A connection is defined by a host, the port number of a Oracle GoldenGate Veridata Agent on that host, and the datasource that is accessed by the agent.
- Configure groups: You need to configure at least one compare group that is linked to a set of source and target datasource connections. A group is a logical container for organizing the objects to be compared.
- Configure compare pairs: You need to configure one or more compare pairs for each group that you create. A compare pair is a set of corresponding source and target tables or files. As part of configuring compare pairs, you map source and target columns to establish a structural relationship between the two objects.
- Configure profiles: A profile contains settings for runtime parameters and can be applied globally to a job, as well as to a specific compare pair as an override to the job profile. Profile parameters encompass considerations such as sorting method, thread and memory usage, reporting output, and so forth. Defining run profiles is optional, because Oracle GoldenGate Veridata includes a default profile that contains settings that apply to most usage scenarios. However, as you gain experience with Oracle GoldenGate Veridata, you may want to customize the default profile or create your own custom profiles.
- Configure jobs: A job is a logical container for one or more compare groups and is the unit of work by which comparison processing is executed. Within one or more jobs, you can manage and run large volumes of compare groups across numerous databases and systems, and you can control the timing of those comparisons.

For more information, see the Oracle GoldenGate Veridata online help.

How Oracle GoldenGate Veridata compares data

Comparison activities are divided into the following steps. You can change some of the aspects of these steps by making parameter changes in Oracle GoldenGate Veridata Web.

Initial comparison step

In the *initial comparison* (or *row hash*) step, rows are retrieved from the source and target tables with a query. If the source and target databases are of different types, the columns are converted to a standardized data type format for accurate comparison. By default, Oracle GoldenGate Veridata compares rows by comparing all columns of the primary key literally (value-for-value) and by using a hash value for all non-key columns. The unique digital signature that is used to calculate the hash value shrinks the data to be transferred over the network for the comparison, while still providing a highly reliable (but not absolute) and efficient mechanism for determining whether two rows contain the same or different column values.

For more assurance of discovering out-of-sync rows, you can configure Oracle GoldenGate Veridata to compare non-key rows column-for-column, instead of using a hash. Full-column comparisons reduce the processing performance in proportion to the number of columns, and they increase network usage.

On the NonStop platform, you can use the delta processing feature during the initial comparison step, if your organization has a license to use server-side sorting. Delta processing is a performance feature by which Oracle GoldenGate Veridata detects which data blocks in the database were modified since a previous comparison and only compares the rows in those blocks. Rows in unchanged blocks are skipped. The default is to compare all rows regardless of whether they changed or not.

There are two steps to delta processing:

- Collecting the base modification time of the previous run for subsequent delta comparisons. This step is always done when delta processing is enabled for a compare pair.
- Comparing data that has been modified since the base comparison, using the information that was collected in the first step. This step is controlled by the "Delta processing enabled" controls on the Compare Pair Configuration page and the Run/Execute Job page of Oracle GoldenGate Veridata Web. These controls allow you to disable the delta comparison step in case there were modifications, such as table reorganizations, that can invalidate the collected delta base information.

For more information about delta processing, see the Oracle GoldenGate Veridata online help.

After the initial comparison, rows that appear to be out-of-sync are stored in a maybe out-of-sync (MOOS) queue in memory, because at this point the comparison is inconclusive. When replication is working concurrently with a comparison, especially if there is replication latency, rows can appear to be out-of-sync when, in fact, the current data is *in flight* (somewhere in the replication flow) and replication will soon synchronize them again.

Confirmation step

The *confirmation*, or *confirm-out-of-sync (COOS)*, step ensures accurate results by confirming row status in a changing environment. In this step, rows are extracted from the MOOS queue, and their status is evaluated as one of the following:

- *in-flight*: the row was out-of-sync in the initial comparison step, but has since been updated. In this case, it is assumed that replication or another mechanism applied the change, but Oracle GoldenGate Veridata was unable to confirm that the rows were in-sync.
- *in-sync*: the source row values were applied to the target row by replication or another method. Even a status of in-sync does not guarantee that the rows are synchronized at any particular moment if the underlying tables are continuously changing, but it does indicate that replication is working.
- *persistently out of sync*: the row has not been updated since the initial comparison step took place, and therefore can be assumed to be out-of-sync.

By default, confirmation processing occurs in a thread that is parallel to the initial comparison step, but the confirmation of each row waits until after a specified replication latency threshold has expired. For example, if latency is 60 seconds, and the initial comparison step revealed an out-of-sync row at 9:30, then the confirmation step for that row is not performed until 9:31 to allow replication to apply any change that was in-flight. After latency is accounted for, rows can be confirmed as persistently out-of-sync and are stored in one or more out-of-sync reports.

Viewing comparison results

Upon completion of a job, you can view the comparison reports and the out-of-sync report by using Oracle GoldenGate Veridata Web or by viewing the files themselves.

Out-of-sync report

You have the option to store an out-of-sync report in binary format, in XML format, or both (or none).

- **OOS file**: When stored in binary form, the OOS report is compatible with the Oracle GoldenGate Veridata programs. As such, it can be viewed from the Oracle GoldenGate Veridata Web interface, and it also can be used as the basis for re-comparing rows that were out of synchronization. To re-compare rows, you select run options to execute another confirmation step, which compares the current state of just those rows and then reports which ones remain out-of-sync after replication or another restorative procedure has been applied.
- **OOSXML file**: When stored as XML, the OOS report is written to an OOSXML file and is stored in a structured way that conforms to an internal XML schema. XML has many advantages, the largest being that it can be manipulated easily by many tools. In its XML form, the file contains all of the information, including metadata, that is needed to select rows for resynchronization by external programs.

Comparison report

Each finished job, group, and compare pair generates a comparison report. The report file contains details about the comparisons that were performed, such as:

- Comparison parameters used
- The number of rows compared and out-of-sync
- The timing of the comparison
- Performance statistics
- Source and target data values

The files themselves are stored as follows:

The OOS files are located in sub-directories of the Oracle GoldenGate Veridata Server installation directory:

- OOS files: shared/data/oos
- OOSXML files: shared/data/oosxml

These directories are further organized by run ID, job name, group name, and compare pair. In the OOSXML directory, the files with the .oosxml extension are the control files. The files with sequential file extensions are the OOSXML chunks. The XML data is spread into multiple files (called “chunks”) for performance purposes.

CHAPTER 2

Oracle GoldenGate Veridata System Requirements

General information

Oracle GoldenGate Veridata does not require an installation of the Oracle GoldenGate replication software. If you do use that software, install Oracle GoldenGate Veridata in a different location.

Supported platforms

Oracle GoldenGate Veridata supports the following databases for comparisons:

- DB2 LUW and z/OS
- Enscribe
- Oracle
- NonStop SQL/MP
- SQL Server
- Teradata

A list of currently supported database versions and operating systems can be found at http://support.oracle.com.

Oracle GoldenGate Veridata Agent system requirements

One Oracle GoldenGate Veridata Agent must be installed for each database instance that contains data that is to be compared. At minimum, therefore, you will install two agents — one to retrieve source rows and one to retrieve target rows (unless you are comparing data within the same database instance).

One agent can retrieve rows from multiple databases or schemas within a given database instance. However, one agent cannot retrieve rows from different database instances.

Oracle GoldenGate Veridata Java agent

Oracle GoldenGate Veridata provides a Java-based agent for the following databases:

- DB2 LUW and Z/OS
- Oracle
- SQL Server
- Teradata

The Java agent connects to the database by using JDBC (Java Database Connectivity). The Java agent enables Oracle GoldenGate Veridata to support comparisons in a heterogeneous environment, where different kinds of databases contain similar, but not identical data types. With a Java agent, one executable supports many operating systems and databases.

You can install the Java agent on the same system as the one where the database is running, or you can install it on a different system, such as the one where Oracle GoldenGate Veridata Server is installed. When considering where to install the agent, weigh the additional and significant use of network bandwidth that will be incurred if the agent is remote from the database, versus the savings in processing resources on the database host when an agent is not running there.

Table 1 Environment requirements to use a Java agent

UNIX and Linux, all supported databases
A Java environment is required on UNIX and Linux systems. Download and install either the Java Software Developer's Kit (JSDK) or the Java Runtime Environment (JRE) from the Sun website. The minimum java version is 1.5. Java 6 is preferred.
DB2
A TCP/IP port must be configured and active.
Oracle
Make certain the listener is configured and running.
SQL Server
A static TCP/IP port must be configured and enabled.
Teradata
◆ Know the host name and port number of the database. ◆ Download the appropriate JDBC driver from the database vendor's website before installing the Java agent. You will be prompted for the location during the installation steps in this guide.

Oracle GoldenGate Veridata C-agent

Oracle GoldenGate Veridata also provides a C-code based agent for the following databases:

- The C-agent is required for NonStop SQL/MP and Enscribe databases running on the NonStop platform. This agent can be installed in a Guardian environment.

- The C-agent is optional for the Oracle database. The C-agent may provide better performance when running like-to-like comparisons.
- You must install the C-agent on the same system as the one that hosts the database.
- For Oracle, a dynamically linked C-agent is available. If you use this agent, you must set the following environment variables:

Table 2 Environment variables for dynamically linked C-agents

Variable	Operating system
PATH=<Oracle libraries>	Microsoft Windows
SHLIB_PATH=<Oracle libraries>	HP-UX
LIBPATH=<Oracle libraries>	IBM AIX
LD_LIBRARY_PATH=<Oracle libraries>	All other supported UNIX platforms

Disk and memory requirements for the agent component

- The agent requires at least 1GB of RAM.
- The disk space requirements for the Oracle GoldenGate Veridata Agent vary by platform, but up to 200 MB may be required. On UNIX and Linux, additional space might be required to install the Java environment (if not already installed).
- The main consumers of processing resources are the row sorting operations that are required during a comparison. To improve performance, you might need to increase the temporary memory space in the database if the columns that are being used as keys are not a native unique index or primary key. You specify the columns to use as keys when configuring Oracle GoldenGate Veridata.
- Using server-side sorting instead of database sorting might reduce the load on the database server and improve comparison performance, depending on the number of rows, the indexes defined, the keys used, and the way the database is tuned. See “Disk and memory requirements for the server component” on page 16.

Database privileges for the agent component

Oracle GoldenGate Veridata Agent makes use of a database login, which must be created before you can run comparisons. You provide the login and password when you configure connection objects in the Oracle GoldenGate Veridata Web interface. The following are the database privileges that are required for the database user.

Table 3 Required database privileges for Oracle GoldenGate Veridata Agent

DB2
◆ SELECT privileges on the tables that will be compared.
Oracle
◆ GRANT CONNECT ◆ GRANT SELECT on the tables to be compared. It is recommended, but not necessary, to GRANT SELECT ANY TABLE. ◆ SELECT_CATALOG_ROLE
NonStop SQL/MP and Enscribe
◆ Read access to the SQL/MP system catalog (for queries to CATALOGS table). ◆ Read access to the SQL/MP catalogs that you want Oracle GoldenGate Veridata to use. ◆ Read access to the DDL dictionaries that you want Oracle GoldenGate Veridata to use. ◆ Read access to the Enscribe and SQL/MP tables that will be compared. ◆ Read, write, create, purge permissions for the Oracle GoldenGate Veridata report and trace files, and access to the subvolumes where they are installed.
SQL Server
◆ db_datareader or the equivalent on the tables to be compared. ◆ VIEW DEFINITION in the databases to be compared. ◆ The database must allow SQL Server authentication.
Teradata
◆ SELECT privileges on the tables to be compared.

Oracle GoldenGate Veridata Server system requirements

Location for the server component

The server and web components are installed from one installation program on Windows, UNIX and Linux systems. The installer includes all files that are needed to run those programs. One installation can be used for comparisons among all of the supported databases, but multiple installations can be used as needed.

Do not install the server or web components on a NonStop system. To use Oracle GoldenGate Veridata for NonStop databases:

- Install the server and web components on a supported Windows, UNIX, or Linux system.
- Make certain that this system has access over high-speed network connections to the NonStop systems.

Additional programs for the server component

Oracle GoldenGate Veridata Server requires a local installation of the Microsoft Visual C++ 2005 Redistributable package. This package is installed by the installer and should be reinstalled with each new Oracle GoldenGate Veridata version. To use an existing package, it must be at least as current as the version that is distributed with Oracle GoldenGate Veridata.

Disk and memory requirements for the server component

The server component uses about 30 MB of fixed virtual memory for basic tasks. The remaining virtual memory is used for comparisons. The main consumers of processing resources on the Oracle GoldenGate Veridata Server machine are the row sorting operations of the NSort component of the initial comparison step when using server-side sorting. Nsort is a third-party sort/merge library that is built into Oracle GoldenGate Veridata.

Enough combined disk space and virtual memory is needed to store all of the rows that are sent for comparison from the source and target systems. To estimate the amount of memory per row:

((number of columns in key + 1) * 4) + 14 + (comparison width of each key column)

Where: comparison width of a key column depends on the comparison format that is selected by Oracle GoldenGate Veridata (or a user override) to use for a comparison.

For example, the number 109998877, if compared as a decimal float, would require:

((1 + 1) * 4) + 14 + 8 = 30 bytes of memory for this row

NOTE This assumes that all non-key columns are compared by using a hash, not literally. More space is needed for literal comparisons.

Beyond this allocation, memory is required for storing rows during the second step of processing, the confirmation step. This can be significant if you expect a large number of rows to require confirmation, as is usually the case when replication latency is very high. These rows are staged in the main memory before they are confirmed.

NSort performs better if all of the data that needs to be sorted during the initial comparison step can be stored in main memory. On 64-bit systems, more memory can be addressed, so more data can be stored in main memory instead of on slower disk devices. The memory that is used in the initial comparison step is not necessarily all released at once to be available for the confirmation step. Consequently, some memory will be shared between processes. When Nsort cannot hold all of the rows in memory, it uses disk storage.

When deciding how much memory to allocate, be aware of the following ways that you can manage it with parameter settings within the Oracle GoldenGate Veridata application:

- You can increase disk I/O performance by specifying multiple temporary directories with profile settings. For maximum benefit, put the directories on different physical disks.
- You can allocate the amount of memory that is allocated to each of the processing steps with profile settings.
- You can use a profile setting to terminate the confirmation step after a given number of out-of-sync rows, to work around resource limitations.
- Additional memory properties can be controlled with server parameters. See page 64.

Repository requirements for the server component

Oracle GoldenGate Veridata Server requires a database to serve as a repository for objects that store the information and environment preferences that users create when using Oracle GoldenGate Veridata Web. You can use the following databases as a repository:

- MySQL
- Oracle
- SQL Server

When installing Oracle GoldenGate Veridata Server, you can have the installer create a database (or schema if using Oracle) for the repository or install the repository objects into an existing database or schema.

The repository can be installed in a database on the same system as the server and web components, or it can be installed on a different system in a multi-tier configuration. For a multi-tier installation, enable communication between hosts as follows:

- To use a MySQL database repository, all required MySQL client software is installed with the server and web components.
- To use an Oracle database repository with Oracle GoldenGate Veridata server and web components installed on a Windows, UNIX, or Linux system, the Oracle client software is installed with the server and web components. Oracle GoldenGate Veridata supports TNSNAMES and EZCONNECT connection naming methods.
    - To use TNSNAMES, the database instance where the repository will be installed must be included in the tnsnames.ora file and in the listener.ora file. Make certain that TNSNAMES is one of the values in the NAMES.DIRECTORY_PATH parameter in the sqlnet.ora file on the database system.
    - To use EZCONNECT, you will need to supply the installation program with the name or IP address of the database server (domain-qualified if required by the operating system), the database listening port number, and the instance name of the database.
- To use a SQL Server database repository with server and web components installed on a different system, make certain that both systems are registered with the DNS.

Database privileges

During installation, you will be prompted for a user (or login) and password for Oracle GoldenGate Veridata to use when updating the repository. You can create the user with the required privileges before running the installation program, or you can allow the installation program to create it with the required privileges. Dedicate this user to Oracle GoldenGate Veridata.

- To have the Oracle GoldenGate Veridata installation program create this user, an Administrator or DBA user and password are required. This user will only be used for this purpose and will not be retained in the Oracle GoldenGate Veridata configuration.
- If creating the user outside the installer, grant the following privileges to create, update, and delete the tables in the repository. Have the user information available for the installation.

Table 4 Required database privileges for Oracle GoldenGate Veridata Server

MySQL

1. Create a user and identically named database.
2. Grant the user all DDL and DML privileges in that database when this user connects to the MySQL server from the host where Oracle GoldenGate Veridata is being installed.

Oracle

1. Create a user and password.
2. Create a database role named VERIDATA_ROLE.
3. Grant these to VERIDATA_ROLE:

   CREATE SESSION
   CREATE TABLE
   CREATE VIEW
   CREATE PROCEDURE
   CREATE SYNONYM
4. Grant VERIDATA_ROLE to the user.
5. Specify QUOTA UNLIMITED on the user's default tablespace.

SQL Server

1. Create a database and login that is a user of the database where you will be installing the repository.
2. Grant this login sufficient privileges to perform the following operations:

   CONNECT to the database

   CREATE, ALTER, DROP TABLE in the database

   CREATE and DROP INDEX in the database

   INSERT, UPDATE, DELETE from tables in the database

   SELECT from tables in the database

   ALTER SCHEMA of the login's default schema

Database environment

During installation, you will be prompted for the following components. Have this information available for the installation.

Table 5 Required environment components for Oracle GoldenGate Veridata Server

MySQL

1. The database host name and port number.
2. A login and password, if using an existing database user for the repository.

Table 5 Required environment components for Oracle GoldenGate Veridata Server (continued)

Oracle

1. The database instance name
2. The ORACLE_HOME (if using TNSNAMES connection method)
3. Default and temporary tablespaces for the repository (if you will be creating a new user for the server component during installation)
4. The host name, port number, and service name of the database instance (if using EZCONNECT connection method)
5. A login and password, if using an existing database user for the repository.

SQL Server

1. An ODBC data source name (DSN). This must be a *system* DSN. You can create the DSN before installing Oracle GoldenGate Veridata or during the installation (you will be prompted for it). A system administrator can create a DSN from the **Data Sources (ODBC)** applet of the **Administrative Tools** control panel. (For basic instructions, see page 77.) Any of the SQL Server clients can be used.
2. The SQL Server instance must be configured to use a static port number. This port is stored in a configuration file so that it can be referenced to make JDBC connections to the database.
3. A login and password (or integrated authentication), if using an existing database user for the repository.

Oracle GoldenGate Veridata Web system requirements

Location for the web component

See "Location for the server component" on page 15. One installer program installs both Oracle GoldenGate Veridata Server and Oracle GoldenGate Veridata Web.

Java environment for the web component

The installation package includes the required Java Runtime Environment (JRE) 1.5.x, which is installed into the JRE directory within the main Oracle GoldenGate Veridata Server installation. To ensure that the web client runs properly, do not change this JRE environment.

Supported browsers for the web component

- Microsoft Internet Explorer version 6 or higher
- Mozilla Firefox version 1.0.4 and higher

Recommended screen resolution for the web component

Minimum of 1024 x 768

Port numbers for the web component

- The Oracle GoldenGate Veridata Web application interacts with the Apache Tomcat web server. Tomcat is installed with Oracle GoldenGate Veridata Server on default ports. You can change the port numbers if needed (see Appendix 1 on page 76).
- Oracle GoldenGate Veridata Web makes use of the following default ports, which you can change during installation:
  - 8820: Shutdown port
  - 8830: HTTP port
  - 4150: Server communication port (requires a password for server-client communication)
  - Other port numbers are used by default. You can change these port numbers after installation by following the directions in “Changing Oracle GoldenGate Veridata Web port numbers” on page 76.

Security for the web component

The Tomcat web server provides user security roles that control access to certain pages of the Oracle GoldenGate Veridata Web interface. You will be asked to specify an initial Oracle GoldenGate Veridata Administrator user and password during installation. You can use this user to create additional users after installation and assign them to roles.

SQL/MP data exclusions

On SQL/MP tables with clustered keys, the right-most portion of the primary key is a timestamp added by the file system. For compare pairs composed of such tables, the timestamp will be different on the source and target systems. As a result, it must be excluded from the comparison, because Oracle GoldenGate Veridata compares keys to ensure that the correct rows are compared.

If, after the timestamp portion of a key is eliminated, the remaining key columns do not produce a unique key value, then those tables must have a unique index or other columns that can be specified to create a unique value. Otherwise, they cannot be compared with Oracle GoldenGate Veridata.

CHAPTER 3

Installing Oracle GoldenGate Veridata Java Agent

These instructions are for installing a new, clean copy of the Java-based Oracle GoldenGate Veridata Agent software. To upgrade an existing installation, see Chapter 12 on page 69.

You can install Oracle GoldenGate Veridata Agent to work with any of the following databases:

- DB2 LUW and z/OS
- Oracle
- SQL Server
- Teradata

The Java agent can be installed on the database host or on a remote host that has network connectivity to the database host. The agent connects by JDBC (Java Database Connectivity).

Downloading the Oracle GoldenGate Veridata files

1. Navigate to http://edelivery.oracle.com.
2. On the **Welcome** page:
   - Select your language.
   - Click **Continue**.
3. On the **Export Validation** page:
   - Enter your identification information.
   - Accept the **Trial License Agreement** (even if you have a permanent license).
   - Accept the **Export Restrictions**.
   - Click **Continue**.
4. On the **Media Pack Search** page:
   - Select the **Oracle Fusion Middleware** Product Pack.
   - Select the platform on which you will be installing the software.

- Click **Go**.

5. In the **Results List**:
   - Select the Oracle GoldenGate Media Pack that you want.
   - Click **Continue**.

6. On the **Download** page:
   - Click **Download** for each component that you want. Follow the automatic download process to transfer the mediapack.zip file to your system.

     NOTE Before installing the software, review the release notes for any new features, new requirements, or bug fixes that affect your current configuration.

Installing the Java agent on a UNIX or Linux system

NOTE Make certain you have reviewed the System Requirements before installing Oracle GoldenGate Veridata Agent.

1. Create a directory for the agent.
2. Extract the mediapack.zip file to the UNIX or Linux system and directory where you want the software to be installed.
3. In the agent directory within the installation directory, make a copy of the agent.properties.sample file, and rename the copy to agent.properties.
4. Open the agent.properties file with a text editor. This file contains comments that help guide your work.
5. Edit the file to provide values for the following parameters:

Table 6 agent.properties file parameters

Parameter	Value
`server.port`	A local port where the agent listens for communication requests from Oracle GoldenGate Veridata Server.
`database.url`	The standard JDBC connection URL string for the database. Samples of the correct format to use are listed with comments in the properties file. You must supply the following values that are appropriate for your database: ◆ Name of the host that contains the database. To use Oracle OCI bequeath protocol, the Oracle OCI libraries must be available locally. ◆ Port number that the database uses. Name of the database or (for Oracle) the instance.This is a sample URL for Oracle. database.url=jdbc:oracle:thin:@localhost:1521:orcl

Table 6 agent.properties file parameters (continued)

Parameter	Value
`server.driversLocation`	Specify the location of the JDBC driver. The JDBC drivers are installed by default into the drivers directory. Change this value only if you need to move the location (not recommended).
`server.jdbcDriver`	The name of the JDBC driver. The drivers are listed with comments in the properties file. Uncomment the correct one for the database. Example (in this case Oracle): `server.jdbcDriver=ojdbc5.jar`
`database.transaction.isolation`	Controls the transaction isolation level that is used during the initial comparison step.

6. Save and close the file.

Installing the Java agent on a Windows system

NOTE Make certain you have reviewed the System Requirements before installing Oracle GoldenGate Veridata Agent.

1. Start the database for which you are installing this agent.
2. Unzip the .zip file that you downloaded.
3. Run the GoldenGate_Veridata_Agent_<platform><version>.exe executable in either of these ways:
   - **As a GUI wizard**: Double click the program and then following these steps. This is the method shown in this documentation.
   - **As a shell process**: From the command console, issue the following command, and then continue with these steps. The prompts are similar to those shown on the wizard screens.

     ```
     Start /wait GoldenGate_Veridata_Agent_<platform><version>.exe -c
     ```
4. **Welcome screen**: Click **Next** on the first page to start the installation.
5. **Destination Directory**: Select an installation directory. You can accept the default (to be created during installation), type a path name, or click **Browse** to go to a different directory. If installing more than one instance of this software, select a different directory or sub-directory each time that you run the installer.
6. **Start Menu Folder**: To add Oracle GoldenGate Veridata Agent to the **Programs** list of the **Start** menu, you can accept the default shortcut name (to be created during installation), type a different one, or select one from the list.
   - To make the shortcut available to everyone who uses this system, select **Create Shortcuts for all users**; otherwise, it will only be available to the current user.
   - To exclude Oracle GoldenGate Veridata Agent from the **Start** menu, select **Don't create a Start Menu folder**. If you are installing this software as a service, it can be started from the installation folder or the **Services** control panel.

7. **Agent Port:** Specify a local port for the agent to use for communication with Oracle GoldenGate Veridata Server.

8. **Database Details**: Select the database type, and specify the name of the host system, the port number that is used by the database, the name of the database (if applicable), and (if Oracle) the name of the instance to which the agent will connect. Under JDBC URL, the resultant connection string is displayed.

9. **Teradata JDBC Drivers**: (Teradata only) Supply the location of the Teradata JDBC driver.

10. **Database Connection Test**: If you kept the default of **Test Database Connection**, a test screen prompts for authentication credentials that you created for the agent to use with this database or for an ODBC driver location, depending on the database type. Click **Test Connection**. Any connection errors are displayed in the output area.

11. **Service Option**: It is recommended that you install Oracle GoldenGate Veridata Agent as a Windows service. When installed as a service, software can operate independently of user connections, which can accidentally be terminated, and you can configure it to start manually or at system start-up. Either accept the default service name or change it as required. If there are, or will be, other instances of this software on the local system, make certain that each name is unique. To prevent installation as a service, deselect the check box.

12. **Start Veridata Agent After Install**: Specify whether or not to start the software after the installation is finished. By default, it is installed to start manually. If you are installing it as a service, a system administrator can change the properties so that it starts automatically when the system starts.

13. **Information**: Review your installation selections. Click **Back** to make changes or click **Next** to begin the installation.

14. Click **Finish** to exit the installer.

CHAPTER 4

Installing Oracle GoldenGate Veridata C-Agent

These instructions are for installing a new, clean copy of the C-based Oracle GoldenGate Veridata Agent software. To upgrade an existing installation, see Chapter 12 on page 69.

The Oracle GoldenGate Veridata C-agent is installed on the same system that hosts the database that contains compare data. You will install one Oracle GoldenGate Veridata C-Agent for each database instance where there is data that is to be compared.

Downloading the Oracle GoldenGate Veridata files

1. Navigate to http://edelivery.oracle.com.
2. On the **Welcome** page:
   - Select your language.
   - Click **Continue**.
3. On the **Export Validation** page:
   - Enter your identification information.
   - Accept the **Trial License Agreement** (even if you have a permanent license).
   - Accept the **Export Restrictions**.
   - Click **Continue**.
4. On the **Media Pack Search** page:
   - Select the **Oracle Fusion Middleware** Product Pack.
   - Select the platform on which you will be installing the software.
   - Click **Go**.
5. In the **Results List**:
   - Select the Oracle GoldenGate Media Pack that you want.
   - Click **Continue**.
6. On the **Download** page:
   - Click **Download** for each component that you want. Follow the automatic download process to transfer the mediapack.zip file to your system.

     NOTE Before installing the software, review the release notes for any new features, new requirements, or bug fixes that affect your current configuration.

Installing the C-agent on a UNIX or Linux system

1. Create a directory for the agent.
2. See See "Downloading the Oracle GoldenGate Veridata files" on page 25 to obtain the software.
3. Extract the mediapack.zip file to directory that you created.
4. From the agent subdirectory, run GGSCI.
5. In GGSCI, issue the following command to create the working directories for Oracle GoldenGate Veridata.

```
CREATE SUBDIRS
```

6. In GGSCI, issue the following command to create and edit a Manager parameter file.

```
EDIT PARAMS MGR
```

7. On the first line of the file, add the following parameter to specify a port number for the Manager process. This port number must be a unique number that is not being used by any other process, including any Manager processes for other Oracle GoldenGate software.

```
PORT <number>
```

8. (Optional) On the next line, add the following parameter to specify a range of up to 256 ports that the Manager process can allocate dynamically. This parameter can be used to specify ports for concurrent processing threads if you will be running batch comparisons.

```
DYNAMICPORTLIST {<port> | <port>-<port>} [ , ...]
```

   ❍ To specify multiple ports, use a comma-delimited list, for example `7830, 7833`.
   ❍ To specify a range of ports, use a dash (-) to separate the first and last port in the range, for example `7830-7835`.
   ❍ To specify a range of ports plus an individual port, place a comma between the range and the individual port number, for example `7830-7835, 7839`.

9. Save and close the parameter file. This file is stored in the dirprm directory within the agent subdirectory. Do not move it.
10. In GGSCI, start the Manager process. You can defer this step until you are ready to run comparisons. To perform comparisons, Manager must be running.

```
START MANAGER
```

11. To confirm that Manager is running, issue the following command in GGSCI.

```
INFO MGR
```

NOTE Oracle GoldenGate Veridata cannot be installed as a service on a UNIX or Linux system.

Installing the C-agent on a Windows system

To install the agent on a Windows system, the following steps are required:

- Install the Oracle GoldenGate Veridata Agent files.
- Specify a custom Manager name (if installed as a service)
- Install the Manager process as a Windows service (recommended).
- Configure the Manager process.

Installing the Oracle GoldenGate Veridata Agent files

1. Create a directory for the agent.
2. See See "Downloading the Oracle GoldenGate Veridata files" on page 25 to obtain the software.
3. Using WinZip or an equivalent compression product, extract the files from the mediapack.zip file to the directory that you created, and make certain to preserve folder names. The extraction creates the installation directory and a subdirectory named agent that contains the software files.
4. From the agent subdirectory, run GGSCI.
5. In GGSCI, issue the following command to create the working directories for the agent.

```
CREATE SUBDIRS
```

6. If you will be installing the Manager as a Windows service and there are other local Manager services, continue with the following steps to specify a custom Manager name. Otherwise, skip that section and follow the instructions for installing Manager as a service on page 28.

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 Oracle GoldenGate replication software and one for Oracle 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.

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 7 INSTALL options

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

Table 7 INSTALL options (continued)

Option	Description
`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.

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

4. (Windows Server 2008) If Windows User Account Control (UAC) is enabled, you are prompted to allow or deny the program access to the computer. Select **Allow** to enable the install program to run. This installs the Manager service with a local system account running with administrator privileges. No further UAC prompts will be encountered when running Manager if installed as a service.

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

Configuring Manager

1. In GGSCI, issue the following command to create and edit a Manager parameter file.

```
EDIT PARAMS MGR
```

2. On the first line of the file, add the following parameter to specify a port number for the Manager process. This port number must be a unique number that is not being used by any other process, including any Manager processes for other Oracle GoldenGate software.

```
PORT <number>
```

3. (Optional) On the next line, add the following parameter to specify a range of up to 256 ports that the Manager process can allocate dynamically. This parameter can be used to specify ports for concurrent processing threads if you will be running batch comparisons.

```
DYNAMICPORTLIST {<port> | <port>-<port>} [ , ...]
```

Where:

- To specify multiple ports, use a comma-delimited list, for example `7830, 7833`.
- To specify a range of ports, use a dash (-) to separate the first and last port in the range, for example `7830-7835`.
- To specify a range of ports plus an individual port, place a comma between the range and the individual port number, for example `7830-7835, 7839`.

4. Save and close the parameter file. This file is stored in the dirprm subdirectory of the agent directory. Do not move it.

5. In GGSCI, start the Manager process. You can defer this step until you are ready to run comparisons. To perform comparisons, Manager must be running.

   ```
   START MANAGER
   ```

6. To confirm that Manager is running, issue the following command in GGSCI.

   ```
   INFO MGR
   ```

Installing the C-agent on a NonStop system

To install the agent on a NonStop SQL/MP system, the following steps are required:

- Install the Oracle GoldenGate Veridata Agent files.
- Copy VSNSERV to remote nodes if they contain table partitions.
- Create a GLOBALS parameter file that contains specifications for:
  - Locations of remote nodes where there is a VSNSERV process.
  - A unique Manager name (if other Manager processes exist on the system)
- Configure the Manager process.

Installing the Oracle GoldenGate Veridata Agent files

1. Download the Oracle GoldenGate Veridata Agent build file to a Windows workstation. See “Downloading the Oracle GoldenGate Veridata files” on page 25.

2. Using WinZip or an equivalent compression product, unzip the files to a temporary directory on your workstation.

3. Transfer the files in binary mode to the volume and subvolume on the NonStop Server where you want to install and run the agent. The agent software must be installed in a dedicated subvolume, including one that is separate from other Oracle GoldenGate software.

4. Alter the VERUNPAK to be an edit file by issuing the following TACL command.

   ```
   FUP ALTER VERUNPAK, CODE 101
   ```

5. Run the VERUNPAK macro by issuing the following TACL command.

   ```
   RUN VERUNPAK
   ```

6. At the prompt, verify the installation location. Type `Y` to confirm the location shown or `N` to select another location.

   ```
   Installing GoldenGate at $DATA.GoldenGate Veridata
   Is this correct? (Y/N) y
   UNPAK - File decompression program - T1255G06 - (2002-05-06)
   Archive version: 1
   File Mode RESTORE Program - T9074G07 (15JAN2002)
   Copyright Tandem Computers Incorporated 1981-2002
   Summary Information
   Files restored = 7  Files not restored = 0
   GoldenGate Veridata for Nonstop Installation
   Installs the GoldenGate Veridata Product
   Enter X at any prompt to quit.
   ```

7. You are prompted for a SQL catalog for the agent to use. Type the catalog name or type `X` for no catalog.

   ```
   SQL Catalog for Compilation (X for no catalog)? $data.cpscat
   SQL compiling VERIAGT
   GoldenGate Veridata Installation Complete.
   ```

8. Continue with "Copying VSNSERV to remote nodes" and "Creating a GLOBALS file" as necessary for your environment.

Copying VSNSERV to remote nodes

If your tables have partitions on remote nodes, you will need to place a copy of the VSNSERV module on each of those nodes.

If all of the remote nodes are the same hardware type, you can use a copy of the VSNSERV that is in the Oracle GoldenGate Veridata agent subvolume. Otherwise, you might need to download the correct agent build for that hardware type. It will include the correct VSNSERV.

To place the VSNSERV on each node, you can do either of the following:

- Install the entire Oracle GoldenGate Veridata Agent package on each of the remote nodes, even though the agent itself will not be running on them.
- Copy the VSNSERV object to each of the remote nodes. To use this option, take the following steps.

To copy VSNSERV to remote nodes

1. Copy the appropriate VSNSERV program to each of the remote nodes.
2. Log onto each remote node as a super user.
3. Issue the following commands on each remote node:

   ```
   FUP GIVE vsnserv, SUPER.SUPER
   FUP secure vsnserv, “NNNN”, PROGID
   ```

   - The first command sets the VSNSERV owner as `SUPER.SUPER`.
   - The second command sets security and PROGID to run as `SUPER.SUPER`.

4. Specify the location of VSNSERV on each remote node by adding a HOST parameter for the node in the GLOBALS file that resides in the Oracle GoldenGate Veridata Agent installation directory. See "Creating a GLOBALS file".

Creating a GLOBALS file

You need to create a GLOBALS file in the Oracle GoldenGate Veridata Agent directory if:

- Other Manager processes exist on this system, such as the one used by Oracle GoldenGate data synchronization software. A unique name for each Manager process must be specified in this file with the GGSPREFIX parameter, including the one that will be used by the Oracle GoldenGate Veridata agent.
- Partitions for tables that will be compared with Oracle GoldenGate Veridata are stored on remote nodes. The name of each node must be specified with the HOST parameter in the GLOBALS file.

To create a GLOBALS file

1. At the TACL prompt, issue the following command.

```
EDIT GLOBALS
```

2. If prompted to create the file, enter Yes.
3. In the GLOBALS file, add one or both of the following parameters, depending on your environment:

```
GGSPREFIX $<aa>
HOST <system name> [, GGSSUBVOL <subvol>] [, NODENUM <node number>]
[HOST <system name> [, GGSSUBVOL <subvol>] [, NODENUM <node number>]]
```

   - GGSPREFIX specifies a unique, two-character prefix that will be attached to the Manager process name, for example GGSPREFIX $GV.
   - HOST specifies the location of remote nodes where there is a VSNSERV component.

NOTE If you do not know the expand node number of a system, run SYSINFO on that node.

```
SYSINFO - T9268H01 - (01 OCT 2004) SYSTEM \TEST Date 10 Jul 2008,
10:44:54
Copyright 2003 Hewlett-Packard Development Company, L.P.

         System name    \TEST
   EXPAND node number    110
        Current SYSnn    SYS10
        System number    012345
  Software release ID    H06.13.00
```

4. Save the file without a file extension. The file is stored in the subvolume where the agent resides. Do not move it.

Configuring Manager

1. From TACL, run the GGSCI program that is installed with the agent.

```
RUN GGSCI
```

2. In GGSCI, issue the following command to create and edit a Manager parameter file.

   ```
   EDIT PARAMS MGRPARM
   ```

3. On the first line of the file, add the following parameter, where `<number>` is a unique port number that is not being used by any other process, including any Manager processes for other Oracle GoldenGate software.

   ```
   PORT <number>
   ```

4. (Optional) On the next line, add the following parameter to specify a range of up to 256 ports that the Manager process can allocate dynamically. You can specify ports for concurrent processing threads if you will be running batch comparisons.

   ```
   DYNAMICPORTLIST {<port> | <port>-<port>} [ , ...]
   ```

   Where:

   - To specify multiple ports, use a comma-delimited list, for example `7830, 7833`.
   - To specify a range of ports, use a dash (-) to separate the first and last port in the range, for example `7830-7835`.
   - To specify a range of ports plus an individual port, place a comma between the range and the individual port number, for example `7830-7835, 7839`.

5. Save and close the file.
6. In GGSCI, issue the following command to start the Manager process. You can defer this step until you are ready to run comparisons. To perform comparisons, Manager must be running.

   ```
   START MANAGER
   ```

7. To confirm that Manager is running, issue the following command in GGSCI.

   ```
   INFO MGR
   ```

CHAPTER 5

Installing the server and web components using a MySQL repository

These instructions are for installing a new, clean copy of the Oracle GoldenGate Veridata Server software for use with a MySQL database repository. To upgrade an existing installation, see Chapter 12 on page 69.

Preinstallation system check

1. Run the operating system's command shell.
2. From the command shell, issue the following command to verify the MySQL installation. If the MySQL software is not installed on the Oracle GoldenGate Veridata host, the command can be executed on the host where MySQL is running, so long as you supply the network name or IP address of that host.

   ```
   mysqladmin -u <admin_user> -h <hostname> -P <port>
   -p<admin_user_password> ping
   ```

   Where:

   - There is no space between the -p and the password.
   - -h <hostname> is always required on UNIX and only required on Windows if the MySQL server is not running on the local machine. If the MySQL server is running on the local machine, use the loopback address of 127.0.0.1, because localhost has special meaning for MySQL and might not work with Oracle GoldenGate Veridata Server.
   - -P <port> is only required when the MySQL server is using a port other than the default of 3306.

3. The following message indicates a correct installation. Any other message indicates a problem that must be resolved before running the installer.

   ```
   mysqld is alive.
   ```

Downloading the Oracle GoldenGate Veridata files

1. Navigate to http://edelivery.oracle.com.
2. On the **Welcome** page:
   - Select your language.

- Click **Continue**.

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

4. On the **Media Pack Search** page:
   - Select the **Oracle Fusion Middleware** Product Pack.
   - Select the platform on which you will be installing the software.
   - Click **Go**.

5. In the **Results List**:
   - Select the Oracle GoldenGate Media Pack that you want.
   - Click **Continue**.

6. On the **Download** page:
   - Click **Download** for each component that you want. Follow the automatic download process to transfer the mediapack.zip file to your system.

     NOTE Before installing the software, review the release notes for any new features, new requirements, or bug fixes that affect your current configuration.

Installing the Oracle GoldenGate Veridata software

1. Start the database that you are using for the Oracle GoldenGate Veridata repository.
2. Close all applications that use this database.
3. Unzip the downloaded .zip file using WinZip or a similar compression program. If installing on a UNIX or Linux system, unzip the file on a Windows system and then FTP the file to the UNIX or Linux system in binary mode.
4. Run the installer.
   - **As a wizard**: Double click the GoldenGate_Veridata_<platform><version> program (Windows) or use a GUI such as X Window (UNIX and Linux) and then follow the installation steps. The Windows wizard is shown in this documentation and contains additional steps not included in the UNIX or Linux verions.
   - **As a shell process**: From the command console of the operating system, issue the following command, and then follow the installation steps. The text prompts are similar to those shown on the wizard screens. To accept the default when using the command console, press **Enter**.

     Windows:

     ```
     Start /wait GoldenGate_Veridata_Agent_<platform><version>.exe -c
     ```

     UNIX and Linux:

     ```
     GoldenGate_Veridata_<platform><version>.sh –c
     ```

5. **Welcome**: Click **Next** to start the installation.

6. **Destination Directory**: Installation Select an installation directory. You can accept the default (to be created during installation), type a path name, or click **Browse** to go to a different directory. If installing more than one instance of this software, select a different directory or sub-directory each time that you run the installer.

7. **Data Location**: Select a location for the user data. The data directory is where Oracle GoldenGate Veridata stores reports and other files that contain information about out-of-sync rows. You can accept the default (to be created during installation), type a path name, or click **Browse** to go to a different directory. This directory will contain data from the tables that are compared, so it should reside on a secure system.

8. **Web Server Configuration**: Accept the default shutdown and HTTP ports or specify different ones. The shutdown port is used by Oracle GoldenGate Veridata to shut down the server processes. The HTTP port is used by the embedded Tomcat web server application to connect to the Oracle GoldenGate Veridata Web component.

9. **Web Server Admin User**: Specify an initial Oracle GoldenGate Veridata administrator user. The administrator is the most privileged role in Oracle GoldenGate Veridata and can perform all configuration, execution, and monitoring functions. Keep a record of this login. It is required to create other users after installation.

10. **Veridata Repository**: Select MySQL as the database that will be the Oracle GoldenGate Veridata repository.

11. **Optional Software**: To download and install a free trial version of MySQL for use as the Oracle GoldenGate Veridata repository, click the link to the MySQL website. Install MySQL and start it. You can leave the installer running while you do this.

12. **Database Driver Configuration**: Specify the name or IP address of the system where you are installing the Oracle GoldenGate Veridata server and web components. Specify the database port number, if it is not the MySQL default port number. If you use **localhost** as the host name, the installer will convert it to the IP address and alert you with a message.

13. **Login Screens**: The next screens vary, depending on your previous selection for the repository configuration.

    - **Veridata Repository Admin**: To create a new user ID or database (or both), you are prompted for the credentials of an existing database user ID that can be used to create the object(s). The user ID will be used only for this purpose and must have administrator or DBA privileges.

    - **Veridata Repository:**

      To create a new user ID, you are prompted to give it a name and password.

      To create a new database, you are prompted to give it a name.

      To use an existing user ID, you are prompted for the login name and password.

      To use an existing database, you are prompted for its name. If the specified database already contains Oracle GoldenGate Veridata objects, you are prompted to specify whether to drop and recreate them (removes existing objects) or to use them (preserves existing objects).

14. **Veridata Repository, database option**: Use any combination of these options to specify how the Oracle GoldenGate Veridata repository will be configured.
    - **User ID**: Oracle GoldenGate Veridata requires a user ID to connect to the repository. Specify whether to **create a new user ID** for this purpose or to **use an existing user ID**. Use an existing user ID if you already created one for Oracle GoldenGate Veridata, or if you are reinstalling Oracle GoldenGate Veridata to a different location from a previous one and want to retain the old repository. An existing user ID must have DML and DDL privileges for the Oracle GoldenGate Veridata repository.
    - **Database**: Specify whether to create a new database for the repository or use an existing one.

15. **Server Configuration**: Specify a listening port for communication between the Oracle GoldenGate Veridata server and web components on this system, or accept the default port number.

16. **Windows Services**: It is recommended that you install the Oracle GoldenGate Veridata server and web components as Windows services. When installed as a service, software can operate independently of user connections, which can accidentally be terminated, and you can configure it to start manually or at system start-up. Either accept the default service names or change them as required. If there are, or will be, other instances of this software on the local system, make certain that each name is unique. To prevent installation as a service, deselect the check box.

17. **Veridata Server Configuration**: Accept the default to install the Microsoft Visual C++ Redistributable Package unless the system contains a version of this package that is at least as current as the one shown by the installer.

18. **Start after install**: Specify whether or not to start the software after the installation is finished. By default, it is installed to start manually. If you are installing it as a service, a system administrator can change the properties so that it starts automatically when the system starts.

19. **Start Menu Folder**: To add Oracle GoldenGate Veridata to the **Programs** list of the **Start** menu, you can accept the default shortcut name (to be created during installation), type a different one, or select one from the list. To make the shortcut available to everyone who uses this system, select **Create Shortcuts for all users**; otherwise, it will only be available to the current user.

    To exclude Oracle GoldenGate Veridata from the **Start** menu, select **Don't create a Start Menu folder**. If you are installing this software as a service, it can be started from the installation folder or the **Services** control panel.

20. **Information**: Review your installation selections. Click **Back** to make changes or click **Next** to begin the installation.

21. **Completing the Oracle GoldenGate Veridata Setup Wizard**: To view the Oracle GoldenGate Veridata help system after the installation is done, accept the default. To prevent the help system from launching, clear the check box. The Oracle GoldenGate Veridata help system contains an introduction to the software, information about specific functions

and tasks, and a detailed tour of the web interface. This tour can help both new and experienced users get started more quickly with a better understanding of the work flow and tools.

22. Click **Finish** to close the installer.

CHAPTER 6

Installing the server and web components using an Oracle repository

These instructions are for installing a new, clean copy of the Oracle GoldenGate Veridata Server software for use with an Oracle database repository. To upgrade an existing installation, see Chapter 12 on page 69.

Preinstallation system check

This check is only required if you are installing the server component to use the TNSNAMES connection method.

1. Run the operating system's command line interface (UNIX/Linux command shell or Windows command console).
2. Execute one of the following commands.
   - *Windows*: Execute outside the Oracle installation bin directory:

     ```
     tnsping <Oracle_instance>
     ```

   - *UNIX/Linux*: Execute as follows:

     ```
     $ORACLE_HOME/bin/tnsping <instance>
     ```

   An OK on the last line of the output confirms that the Oracle installation files and DLLs are installed correctly. Any other result indicates a problem, which must be fixed before proceeding with these steps.
3. After a successful tnsping command, execute one of the following commands as an administrator user.
   - *Windows*:

     ```
     sqlplus <admin_user>/<admin_user_password>@<instance>
     ```

   - *UNIX/Linux*:

     ```
     sqlplus <admin_user>/<admin_user_password>@<instance>
     ```

   The result should be a SQL*Plus command prompt. If you do not receive the SQL*Plus command prompt, it means that you cannot connect to the database by using a connect string. This problem must be fixed before running the installer.

Downloading the Oracle GoldenGate Veridata software

1. Navigate to http://edelivery.oracle.com.
2. On the **Welcome** page:
   - Select your language.
   - Click **Continue**.
3. On the **Export Validation** page:
   - Enter your identification information.
   - Accept the **Trial License Agreement** (even if you have a permanent license).
   - Accept the **Export Restrictions**.
   - Click **Continue**.
4. On the **Media Pack Search** page:
   - Select the **Oracle Fusion Middleware** Product Pack.
   - Select the platform on which you will be installing the software.
   - Click **Go**.
5. In the **Results List**:
   - Select the Oracle GoldenGate Media Pack that you want.
   - Click **Continue**.
6. On the **Download** page:
   - Click **Download** for each component that you want. Follow the automatic download process to transfer the mediapack.zip file to your system.

     NOTE Before installing the software, review the release notes for any new features, new requirements, or bug fixes that affect your current configuration.

Installing the Oracle GoldenGate Veridata software

1. Start the database that you are using for the Oracle GoldenGate Veridata repository.
2. Close all applications that use this database.
3. Unzip the downloaded .zip file using WinZip or a similar compression program. If installing on a UNIX or Linux system, unzip the file on a Windows system and then FTP the file to the UNIX or Linux system in binary mode.
4. Run the installer.
   - **As a wizard**: Double click the GoldenGate_Veridata_<platform><version> program (Windows) or use a GUI such as X Window (UNIX and Linux) and then follow the installation steps. The Windows wizard is shown in this documentation and contains additional steps not included in the UNIX or Linux verions.

- **As a shell process**: From the command console of the operating system, issue the following command, and then follow the installation steps. The text prompts are similar to those shown on the wizard screens. To accept the default when using the command console, press **Enter**.

  Windows:

  ```
  Start /wait GoldenGate_Veridata_Agent_<platform><version>.exe -c
  ```

  UNIX and Linux:

  ```
  GoldenGate_Veridata_<platform><version>.sh –c
  ```

5. **Welcome**: Click **Next** to start the installation.

6. **Destination Directory**: Installation Select an installation directory. You can accept the default (to be created during installation), type a path name, or click **Browse** to go to a different directory. If installing more than one instance of this software, select a different directory or sub-directory each time that you run the installer.

7. **Data Location**: Select a location for the user data. The data directory is where Oracle GoldenGate Veridata stores reports and other files that contain information about out-of-sync rows. You can accept the default (to be created during installation), type a path name, or click **Browse** to go to a different directory. This directory will contain data from the tables that are compared, so it should reside on a secure system.

8. **Web Server Configuration**: Accept the default shutdown and HTTP ports or specify different ones. The shutdown port is used by Oracle GoldenGate Veridata to shut down the server processes. The HTTP port is used by the embedded Tomcat web server application to connect to the Oracle GoldenGate Veridata Web component.

9. **Web Server Admin User**: Specify an initial Oracle GoldenGate Veridata administrator user. The administrator is the most privileged role in Oracle GoldenGate Veridata and can perform all configuration, execution, and monitoring functions. Keep a record of this login. It is required to create other users after installation.

10. **Veridata Repository**: Select Oracle as the database that will be the Oracle GoldenGate Veridata repository.

11. One of the following screens appears:
    - **Oracle Database Home Location**: This screen appears if there is only one Oracle instance installed, or if you are installing on a UNIX system where the ORACLE_HOME environment variable is defined. Verify that this Oracle installation location is the one that belongs to the database that will be the Oracle GoldenGate Veridata repository. If this is not the correct Oracle Home, click **Cancel** to exit the installer. Consult the database documentation for how to set the Oracle Home. After correctly setting the Oracle Home, run the installer again.
    - **Oracle Database Configuration**: If this screen appears, the installer was unable to locate a tnsnames.ora file. You can bypass the tnsnames.ora file by selecting EZCONNECT as the connection method, or you can click **Browse** to point the installer to the correct tnsnames.ora file location.

12. **Database Driver Configuration**: Specify the SQL*Net connection information to use.

NOTE Depending on whether you forced tnsnames or EZCONNECT on the previous screen (if it appeared in your case), you may be permitted to select only one of these options.

- Select **Use TNS names** to have Oracle GoldenGate Veridata get connection information from the tnsnames.ora file. The drop-down list displays all Oracle instances that are listed in the TNSNAMES.ORA file that is associate with the previously selected ORACLE_HOME. Make certain that the entry for this instance in the file is correct and that this same service name is configured in the listener.ora file. Make certain that TNSNAMES is one of the values in the NAMES.DIRECTORY_PATH parameter in the sqlnet.ora file on the database system. Many installation and connection problems can be traced to incorrect connection identifiers in these files.

  NOTE On a UNIX system, if the installer is run from the command line instead of from a GUI, it may not display all of the Oracle instances in TNSNAMES.ORA, but only the first Oracle instance in the list. Type the correct instance name if it is not displayed.

- Select **Use EZCONNECT** to have Oracle GoldenGate Veridata connect by using EZCONNECT. Then, supply the name or IP address of the database server (domain-qualified name, if required by the operating system), the database listening port number (default is 1521), and the instance name of the database. If connection errors occur, make certain that EZCONNECT is specified for the NAMES.DIRECTORY_PATH parameter in the sqlnet.ora file on the database system.

13. **Veridata Repository, database option**: Oracle GoldenGate Veridata requires a database user ID or schema to own and connect to the repository. Specify whether to **create a new user ID** for this purpose or to **use an existing user ID**. Use an existing user ID or schema if you already created one for Oracle GoldenGate Veridata, or if you are reinstalling Oracle GoldenGate Veridata to a different location from a previous one and want to retain the old repository. An existing user ID must have DML and DDL privileges for the Oracle GoldenGate Veridata repository.

14. **Login Screens:** The next screens vary, depending on your previous selection for the repository configuration.

    - **To create a new user**:

      Veridata Repository Admin: Supply the login credentials of an existing database user that has DBA or system administrator privileges to create the Oracle GoldenGate Veridata user and schema.

      Click **Next** and then supply login credentials for the new Oracle GoldenGate Veridata user. The installer creates the new user with the necessary privileges to operate as Oracle GoldenGate Veridata.

    - **To use an existing user**:

      Supply the login credentials of the existing user, and then click **Next**.

If the specified schema already contains Oracle GoldenGate Veridata objects, you are prompted to specify whether to drop and recreate it (removes existing objects) or to use it (preserves existing objects). Click **Next** after making a selection.

15. (New user only) **Veridata Oracle Tablespace**: Specify default and temporary tablespaces for the Oracle GoldenGate Veridata repository.

16. **Server Configuration**: Specify a listening port for communication between the Oracle GoldenGate Veridata server and web components on this system, or accept the default port number.

17. **Windows Services**: It is recommended that you install the Oracle GoldenGate Veridata server and web components as Windows services. When installed as a service, software can operate independently of user connections, which can accidentally be terminated, and you can configure it to start manually or at system start-up. Either accept the default service names or change them as required. If there are, or will be, other instances of this software on the local system, make certain that each name is unique. To prevent installation as a service, deselect the check box.

18. **Veridata Server Configuration**: Accept the default to install the Microsoft Visual C++ Redistributable Package unless the system contains a version of this package that is at least as current as the one shown by the installer.

19. **Start after install**: Specify whether or not to start the software after the installation is finished. By default, it is installed to start manually. If you are installing it as a service, a system administrator can change the properties so that it starts automatically when the system starts.

20. **Start Menu Folder**: To add Oracle GoldenGate Veridata to the **Programs** list of the **Start** menu, you can accept the default shortcut name (to be created during installation), type a different one, or select one from the list. To make the shortcut available to everyone who uses this system, select **Create Shortcuts for all users**; otherwise, it will only be available to the current user.

    To exclude Oracle GoldenGate Veridata from the **Start** menu, select **Don't create a Start Menu folder**. If you are installing this software as a service, it can be started from the installation folder or the **Services** control panel.

21. **Information**: Review your installation selections. Click **Back** to make changes or click **Next** to begin the installation.

22. **Completing the Oracle GoldenGate Veridata Setup Wizard**: To view the Oracle GoldenGate Veridata help system after the installation is done, accept the default. To prevent the help system from launching, clear the check box. The Oracle GoldenGate Veridata help system contains an introduction to the software, information about specific functions and tasks, and a detailed tour of the web interface. This tour can help both new and experienced users get started more quickly with a better understanding of the work flow and tools.

23. Click **Finish** to close the installer.

CHAPTER 7

Installing the server and web components using a SQL Server repository

These instructions are for installing a new, clean copy of the Oracle GoldenGate Veridata Server software for use with a SQL Server database repository. To upgrade an existing installation, see Chapter 12 on page 69.

Preinstallation system check

If you have not done so already, create a system data source (DSN) for Oracle GoldenGate Veridata to use to connect to the SQL Server instance. See page 77. If you have created a DSN already, you can use the same steps to verify the connection.

Downloading the Oracle GoldenGate Veridata software

1. Navigate to http://edelivery.oracle.com.
2. On the **Welcome** page:
   - Select your language.
   - Click **Continue**.
3. On the **Export Validation** page:
   - Enter your identification information.
   - Accept the **Trial License Agreement** (even if you have a permanent license).
   - Accept the **Export Restrictions**.
   - Click **Continue**.
4. On the **Media Pack Search** page:
   - Select the **Oracle Fusion Middleware** Product Pack.
   - Select the platform on which you will be installing the software.
   - Click **Go**.
5. In the **Results List**:
   - Select the Oracle GoldenGate Media Pack that you want.

- Click **Continue**.

6. On the **Download** page:

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

     NOTE Before installing the software, review the release notes for any new features, new requirements, or bug fixes that affect your current configuration.

Installing the Oracle GoldenGate Veridata software

1. Start the database that you are using for the Oracle GoldenGate Veridata repository.
2. Close all applications that use this database.
3. Unzip the downloaded .zip file using WinZip or a similar compression program.
4. Run the installer.
   - **As a wizard**: Double click the GoldenGate_Veridata_<platform><version> program and then follow the installation steps. The wizard is shown in this documentation.
   - **As a shell process**: From the command console of the operating system, issue the following command, and then follow the installation steps. The text prompts are similar to those shown on the wizard screens. To accept the default when using the command console, press **Enter**.

     ```
     Start /wait GoldenGate_Veridata_Agent_<platform><version>.exe -c
     ```

5. **Welcome**: Click **Next** to start the installation.
6. **Destination Directory**: Installation Select an installation directory. You can accept the default (to be created during installation), type a path name, or click **Browse** to go to a different directory. If installing more than one instance of this software, select a different directory or sub-directory each time that you run the installer.
7. **Data Location**: Select a location for the user data. The data directory is where Oracle GoldenGate Veridata stores reports and other files that contain information about out-of-sync rows. You can accept the default (to be created during installation), type a path name, or click **Browse** to go to a different directory. This directory will contain data from the tables that are compared, so it should reside on a secure system.
8. **Web Server Configuration**: Accept the default shutdown and HTTP ports or specify different ones. The shutdown port is used by Oracle GoldenGate Veridata to shut down the server processes. The HTTP port is used by the embedded Tomcat web server application to connect to the Oracle GoldenGate Veridata Web component.
9. **Web Server Admin User**: Specify an initial Oracle GoldenGate Veridata administrator user. The administrator is the most privileged role in Oracle GoldenGate Veridata and can perform all configuration, execution, and monitoring functions. Keep a record of this login. It is required to create other users after installation.
10. **Veridata Repository**: Select SQL Server as the database that will be the Oracle GoldenGate Veridata repository.
11. **Database Driver Configuration**: Specify the correct SQL Server ODBC data source name (DSN) for Oracle GoldenGate Veridata to use to connect to the repository database. If a DSN does not exist, or if no DSNs are displayed on this installer page, you can leave

the installer running while you create the DSN, and then resume the installation afterward. Contact your system administrator or consult the documentation for your operating system if you are unable to create a DSN. For basic instructions, see page 77.

12. Confirm the DSN connection information.
13. **Veridata Repository, database option**: Use any combination of these options to specify how the Oracle GoldenGate Veridata repository will be configured.
    - **Login**: Oracle GoldenGate Veridata requires a login to connect to the repository. Specify whether to **create a new login** for this purpose or to **use an existing login**. Use an existing login if you already created one for Oracle GoldenGate Veridata, or if you are reinstalling Oracle GoldenGate Veridata to a different location from a previous one and want to retain the old repository. An existing login must have DML and DDL privileges for the Oracle GoldenGate Veridata repository.
    - **Database**: Specify whether to create a new database for the repository or use an existing one.
14. **Login Screens**: The next screens vary, depending on your previous selection for the repository configuration.
    - **Veridata Repository Admin**: To create a new login or database (or both), you are prompted for the credentials of an existing database login that can be used to create the object(s). The login will be used only for this purpose and must have administrator or DBA privileges.
    - **Veridata Repository:**

      To create a new login, you are prompted to give it a name and password.

      To create a new database, you are prompted to give it a name.

      To use an existing login, you are prompted for the login name and password.

      To use an existing database, you are prompted for its name. If the specified database already contains Oracle GoldenGate Veridata objects, you are prompted to specify whether to drop and recreate them (removes existing objects) or to use them (preserves existing objects).
15. **Server Configuration**: Specify a listening port for communication between the Oracle GoldenGate Veridata server and web components on this system, or accept the default port number.
16. **Windows Services**: It is recommended that you install the Oracle GoldenGate Veridata server and web components as Windows services. When installed as a service, software can operate independently of user connections, which can accidentally be terminated, and you can configure it to start manually or at system start-up. Either accept the default service names or change them as required. If there are, or will be, other instances of this software on the local system, make certain that each name is unique. To prevent installation as a service, deselect the check box.
17. **Veridata Server Configuration**: Accept the default to install the Microsoft Visual C++ Redistributable Package unless the system contains a version of this package that is at least as current as the one shown by the installer.

18. **Start after install**: Specify whether or not to start the software after the installation is finished. By default, it is installed to start manually. If you are installing it as a service, a system administrator can change the properties so that it starts automatically when the system starts.

19. **Start Menu Folder**: To add Oracle GoldenGate Veridata to the **Programs** list of the **Start** menu, you can accept the default shortcut name (to be created during installation), type a different one, or select one from the list. To make the shortcut available to everyone who uses this system, select **Create Shortcuts for all users**; otherwise, it will only be available to the current user.

    To exclude Oracle GoldenGate Veridata from the **Start** menu, select **Don't create a Start Menu folder**. If you are installing this software as a service, it can be started from the installation folder or the **Services** control panel.

20. **Information**: Review your installation selections. Click **Back** to make changes or click **Next** to begin the installation.

21. **Completing the Oracle GoldenGate Veridata Setup Wizard**: To view the Oracle GoldenGate Veridata help system after the installation is done, accept the default. To prevent the help system from launching, clear the check box. The Oracle GoldenGate Veridata help system contains an introduction to the software, information about specific functions and tasks, and a detailed tour of the web interface. This tour can help both new and experienced users get started more quickly with a better understanding of the work flow and tools.

22. Click **Finish** to close the installer.

23. The SQL Server repository must be running before the Oracle GoldenGate Veridata Server Windows service can start. You can add the database as a startup dependency to the server service. Adding a dependency to a service requires a system administrator to modify its Registry entry. For more information, consult the Microsoft documentation or website.

CHAPTER 8

Configuring Oracle GoldenGate Veridata security

When using Oracle GoldenGate Veridata, you will be selecting, viewing and storing data values from the tables or files of your business applications. Care must be taken to protect access to the following components:

- The files, programs, and directories in the Oracle GoldenGate Veridata installation directories
- Data files that contain the results of data comparisons
- The Oracle GoldenGate Veridata Web interface, where data values can be viewed

Securing the Oracle GoldenGate Veridata files

Controlling access to the installation directories

Standard operating system permissions apply to the programs, files, and directories within the Oracle GoldenGate Veridata Server, Oracle GoldenGate Veridata Web, and Oracle GoldenGate Veridata Agent installation directories. You should adjust the permissions for these objects based on your business security rules.

Securing files that contain business data

Oracle GoldenGate Veridata Server creates data files that will contain sensitive application data. By default, these files reside in the shared/data directory within the Oracle GoldenGate Veridata Server installation directory, but the person who installed the software might have installed them in a different location. All of the sub-directories within that directory contain files that may reflect business data.

The types of files that contain sensitive data are:

- The comparison report (rpt sub-directory)
- The out-of-sync report (oosxml and oos sub-directories)

These files inherit the same file permissions as those of the user that runs the Oracle GoldenGate Veridata Server installation program. Do not change the permissions, or Oracle GoldenGate Veridata may be unable to maintain them. These files should be kept just as secure as you would keep your business data. Users of Oracle GoldenGate Veridata Web do not require access to these files because they see the same information through the client interface.

Securing access to Oracle GoldenGate Veridata Web

You can assign security roles to the users of Oracle GoldenGate Veridata to control their access to the functions that are performed by the software, some of which expose selected data values from the database. These roles are:

- **Administrator**: The administrator role is the highest-level security role in Oracle GoldenGate Veridata. This role can perform all of the functions that configure, execute, and monitor Oracle GoldenGate Veridata.
- **PowerUser**: The power user role is the second-highest role in Oracle GoldenGate Veridata. This role can perform all of the functions that configure, execute, and monitor Oracle GoldenGate Veridata from the Oracle GoldenGate Veridata Web interface, but this role cannot perform any configuration functions for the Oracle GoldenGate Veridata Server.
- **ReportViewer**: The report viewer role cannot perform functions that configure Oracle GoldenGate Veridata or execute jobs. This role can only view configuration and job information, and view comparison reports.
- **DetailReportViewer**: The detail report viewer role cannot perform any functions that configure Oracle GoldenGate Veridata or execute jobs. This role can only view configuration and job information, and view comparison reports and out-of-sync report information through the Oracle GoldenGate Veridata Web interface or at the file level.

Security is controlled through the Administration Tool of the Apache Tomcat Web Server. From this interface, a user with the administrator role can:

- Create a user and assign it a security role.
- Create user groups and assign them security roles. Users can be added to these groups without being given a security role. A user inherits the role of its group.
- Create a user and assign it a security role, and then add that user to a group. The user inherits the role of its group and keeps its individual role.

To open the Apache Tomcat Web Server Administration Tool

1. Connect to the Apache Tomcat Web Server from a browser by typing the following address:

   ```
   http://<hostname>:<port>/admin
   ```

   Where: <hostname> is the name or IP address of the system where the Oracle GoldenGate Veridata server and web components are hosted, and <port> is the port number assigned to Oracle GoldenGate Veridata Server (default is 8830).

2. Log on to the Apache Tomcat Web Server Administration Tool as an Oracle GoldenGate Veridata administrator user. A default administrator user was created during the installation of Oracle GoldenGate Veridata.

3. In the navigation pane, expand **User Definition**. From this node, all user resources are managed.

To create a group

1. Under **User Definition**, click **Groups**. Existing groups are displayed and can be edited by clicking their names.

2. From the **Group Actions** list, select **Create New Group**.

3. Under **Group Properties**, type a name for the group (no spaces, case sensitive) in the **Group Name** box and (optional) a description in the **Description** box.
4. Under **Role Name**, select the check box next to the role you want to assign to the group. You can select any role that is listed in "Securing access to Oracle GoldenGate Veridata Web".
5. Click **Save** to save the group.
6. When finished using the Apache Tomcat Web Server Administration Tool, click **Commit Changes** to save the changes to the repository. To make any more changes after you commit the changes, you must log in again.

To create or edit a user

1. Under **User Definition**, click **Users**.
2. To edit a user, click its name. To add a new user, select **Create New User** from the **User Actions** list.
3. Under **User Properties**, type:
   - **User Name**: A name for the user (no spaces, case-sensitive)
   - **Password**: A password for the user (no spaces, case-sensitive).
   - **Full Name**: (Optional) the name of the person who is this user.
4. To assign the user to a group, click the check box next to the name of the group under **Group Name**. Linking a user to a group is optional. The user inherits the default role of the group.
5. To assign a role to this user, click the check box next to the name of the role under **Role Name**. You can select any role that is listed in "Securing access to Oracle GoldenGate Veridata Web".
6. Click **Save** to save the user.
7. When finished using the Apache Tomcat Web Server Administration Tool, click **Commit Changes** to save the changes to the repository. To make any more changes after you commit the changes, you must log in again.

CHAPTER 9

Running the Oracle GoldenGate Veridata Programs

You can start the Oracle GoldenGate Veridata programs in any order.

Starting and stopping the C-agent and Manager

The C-agent starts automatically at the request of Oracle GoldenGate Veridata Server when initiating comparisons. However, for Oracle GoldenGate Veridata Agent to function correctly, the following must be running:

- The database to which the agent is linked.
- The Manager process for the C-agent.

Although the agent process itself is an automatic process, you can stop the Manager process that controls the agent. Stopping Manager prevents Oracle GoldenGate Veridata Server from being able to start a new agent process, but it does not stop agents that are already running.

To control the C-agent Manager on all platforms

1. From the Oracle GoldenGate Veridata Agent installation location, run GGSCI.
2. In GGSCI, issue the appropriate command as follows to stop or start the Manager.

   ```
   START MANAGER
   ```
   Or...
   ```
   STOP MANAGER
   ```

To control the C-agent Manager as a Windows service

If Manager is installed as a Windows service, it either starts at system boot time or must be started manually, depending on how the service is configured. The installation default is to start automatically at system boot time. To start the service manually, use the START MANAGER command in GGSCI or start the service in the **Services** control panel applet.

To change the startup behavior of the service, right click the name in the **Services** control panel, and then select **Properties**.

When Manager is installed as a Windows service, you can stop it in the **Services** control panel applet or with the STOP MANAGER command in GGSCI.

Starting and stopping the Java-based components

The Oracle GoldenGate Veridata Server and Oracle GoldenGate Veridata Web components are Java-based programs. The Oracle GoldenGate Veridata Agent component is also available as a Java program for all platforms except NonStop.

NOTE Before starting the server and web processes, start the repository database.

Controlling the Java-based components from the command line

To control the agent component, change directories to its installation directory and issue the appropriate command as follows:

UNIX and Linux

```
agent.sh {start | run}
Or...
agent.sh stop
```

Windows

```
agent.bat {start | run}
Or...
agent.bat stop
```

Where:

- run starts the agent in the same command window from which it is launched.
- start starts the agent in a separate command window.

NOTE The run option is useful for diagnosing errors that happen during the startup process before the agent error logging is configured. When the run option is used, messages written stdout and stderr appear in the command window. The agent normally logs its messages to the log file, so only operating system messages and logging system errors are written to stderr. When the start option is used, messages written to stdout and stderr are discarded. .

To control the server component, change directories to the server/bin directory of the installation directory, and issue the appropriate command as follows:

UNIX and Linux

```
veridata_server.sh start
Or...
veridata_server.sh stop
```

Windows

```
veridata_server.bat start
Or...
veridata_server.bat stop
```

To control the web component, change directories to the web/bin directory of the installation directory, and issue the appropriate command as follows:

UNIX and Linux

```
veridata_web.sh start
Or...
veridata_web.sh stop
```

Windows

```
veridata_web.bat start
Or...
veridata_web.bat stop
```

Configure the host to start and stop the processes automatically. Contact your system administrator if you need assistance.

NOTE If you are restarting Oracle GoldenGate Veridata Server, you must also restart Oracle GoldenGate Veridata Web.

Controlling the Java-based components as Windows services

An Oracle GoldenGate Veridata Windows service either starts at system boot time or must be started manually, depending on how it is configured. The installation default is to start automatically at system boot time. Windows services are controlled from the **Services** control panel applet or the **Start** menu. To change the startup behavior of a service, right click the name in the **Services** control panel, and then select **Properties**.

When installed as a service, an Oracle GoldenGate Veridata Java component can be stopped from the **Services** control panel, from the **Start** menu, or from the command console.

To control the Java components from the Windows Start Menu

1. In the **Start** menu, select **Programs**, and then navigate to the folder that contains the Oracle GoldenGate Veridata shortcuts.
   - To start the server component: **Application**>**Start Veridata Server**
   - To stop the server component: **Application**>**Stop Veridata Server**

   - To start the web component: **Application**>**Start Veridata Web**
   - To stop the web component: **Application**>**Stop Veridata Web**.

   - To start the agent component: **Application**>**Start Veridata Agent**
   - To stop the agent component: **Application**>**Stop Veridata Agent**

Reloading log information

You can reload logging information from the log4j. properties file to a running agent. The changes in the log4j. properties file are put into effect on the agent. The agent must be running for this command to work. Use the following command:

UNIX and Linux	Windows
`agent.sh reloadlog`	`agent.bat reloadlog`

Connecting to Oracle GoldenGate Veridata Web

You can connect to Oracle GoldenGate Veridata Web in either of the following ways.

- If there is a **Start** menu shortcut for Oracle GoldenGate Veridata on the system where the server and web components are installed, navigate to it and select **Launch Oracle GoldenGate Veridata Web**.
- Run a web browser on any system in your network and type the following address:

  ```
  http://<hostname>:<port>/veridata
  ```

  Where: <hostname> is the name of the system where Oracle GoldenGate Veridata Server is installed and <port> is the port number where it is running (default is

8830). Use localhost as the host name if connecting on the system that is local to the server installation.

Examples:

```
http://localhost:8830/veridata
http://sysa:8830/veridata
```

The Oracle GoldenGate Veridata Web login page is displayed upon successful connection. Log in with your user name and password. For full instructions on using Oracle GoldenGate Veridata Web, see the online help.

CHAPTER 10

Running comparisons from the command line

Overview of the command line interface

You can use the vericom tool of Oracle GoldenGate Veridata to execute certain comparison tasks from the command shell of the operating system. The vericom tool runs the Oracle GoldenGate Veridata Command Line Interface and enables you to handle these activities with automated programs.

You can:

- Run an entire job or a specific compare pair of a job. **Note:** You cannot run a group individually.
- Shut down Oracle GoldenGate Veridata Server
- Set tracing (only under guidance of an Oracle Support analyst)

For specific compare pairs, you can:

- Review previous out-of-sync results
- Generate out-of-sync XML from the previous run
- Override the same profile and row partition settings that are possible from the web interface

Comparisons also can be run from the Oracle GoldenGate Veridata Web interface. This interface provides greater control in configuring the objects to be compared and for controlling runtime parameter settings.

Running vericom

The vericom program can be run by anyone who has the correct operating system permissions to run it.

1. On the system where the Oracle GoldenGate Veridata Server is installed, run the command shell of the operating system.
2. Change directories to server/bin within the Oracle GoldenGate Veridata Server installation directory.
3. Use the following syntax to run the vericom program.

Syntax

```
vericom -<required input> -<optional input>
```

Required input

One of the following are required; otherwise an error is returned.

```
-shutdown |
-help |
-helprun |
-{version | v} |
-{job | j} <job>
```

If -version, -v, -help, or -helprun are specified, they take precedence over any other flag specified.

Optional input

```
[ -g <group> -c <compare_pair> ]
[ -rP <profile> ]
[ -rR ]
| -rO ]
[ -rN <threads> ]
[ -rD <seconds> ]
[ -rC | +rC ]
[ -rOb | -rOx | -rO2 | -rO0 ]
[ -rOs <records> ]
[ -rTi ]
[ -rTc ]
[ -rTs <trace_number> ]
[ -pS <source_partition_name> |
   -pSq <source_sql_predicate> |
   -pSA1 <source_ascii_start_key> |
   -pSA2 <source_ascii_end_key> |
   -pSH1 <source_hex_start_key> |
   -pSH2 <source_hex_end_key> ]
[ -pT <target_partition_name> |
   -pTq <target_sql_predicate> |
   -pTA1 <target_ascii_start_key> |
   -pTA2 <target_ascii_end_key> |
   -pTH1 <target_hex_start_key> |
   -pTH2 <target_hex_end_key> ]
[ -pq <sql_predicate> ]
[ -w | { -wp <poll_interval> } ]
[ -rd0 | -rdN <run_ID> ]
[ -{url | u} <url> ]
[ -{password | p} <password> ]
```

Table 8 Vericom runtime arguments

Argument	Description
`-shutdown`	Shuts down the Oracle GoldenGate Veridata Server processes.
`-help`	Displays the vericom syntax components and their descriptions.
`-helprun`	Displays run-related syntax components and their descriptions.
`-{version | v}`	Displays the version of the Oracle GoldenGate Veridata command-line interface that is being used.
`-{job | j} <job>`	Specifies the job to be run. For <job>, specify the name that was assigned when the job was created in Oracle GoldenGate Veridata Web.

`-g <group>`
`-c <compare_pair>`
	Specifies a group and compare pair. For <group> and <compare_pair>, specify the names that were assigned when these objects were created in Oracle GoldenGate Veridata Web.
	◆ If -g and -c are used, -j must also be used.

`-rP <profile>`	Overrides the profile that is defined for a job. For <profile>, specify the name that was assigned when the profile was created in Oracle GoldenGate Veridata Web. ◆ If -rP is used, -j must be used.
`-rR`	A run override option. Compares only those rows that were out-of-sync in the previous run, based on the information that is stored in the out-of-sync file. The results identify which rows were brought back into synchronization by replication or another method. ◆ Do not use -rR and -rO in the same run.
`-rO`	A run override option. Generates an OOSXML file based on the out-of-sync file from the previous run. It generates XML for every row that is in the file. You can use the XML to view the out-of-sync information in an XML editor or for other purposes. ◆ -rO must be used with -j. ◆ Do not use -rR and -rO in the same run.

Table 8 Vericom runtime arguments (continued)

Argument	Description
`-rN <threads>`	Specifies the number of concurrent comparison threads to use. You can use as many threads as there are processors on the server system. This option overrides the default job profile and has no effect if a job is not run with -j or if just one comparison is run by using -j with -g and -c.
`-rD <seconds>`	Delays the confirmation step by the specified number of seconds to account for replication lag. Delaying the confirmation step reduces the number of false out-of-sync results that occur because an updated source value was not replicated fast enough. This option overrides the default job profile and has no effect if the -rR option is used.
`-rC | +rC`	Controls whether or not the confirmation step (confirm OOS) is performed in the job. ◆ -rC skips the confirmation step. You can skip the confirmation step if activity on the source tables is stopped or if replication is not continuously updating the target table(s). ◆ +rC includes the confirmation step. These options override the default job profile and are mutually exclusive. They have no effect unless -j is used.
`-rOb | -rOx | -rO2 | -rO0`	Controls the kind of file that is produced for the out-of-sync report. ◆ -rOb generates binary format that is compatible with the Oracle GoldenGate Veridata Web browser. ◆ -rOx generates output in XML. ◆ -rO2 generates both binary and XML output. ◆ -rO0 suppresses out-of-sync output. These options override the default job profile and are mutually exclusive. They have no effect if -rR is used.
`-rOs <records>`	Limits the number of out-of-sync rows that are written to a chunk of the OOSXML file. Writing the file in chunks prevents it from becoming too large for the system to manage and allows periodic archiving or purging. The current file is closed when the specified number of rows is written, and a new file is opened. This option overrides the default job profile and has no effect if -rR is used.
`-rTi`	Turns on tracing of Oracle GoldenGate Veridata Agent for the initial comparison step. Do not use without the guidance of an Oracle support analyst.

Table 8 Vericom runtime arguments (continued)

Argument	Description
`-rTc`	Turns on tracing of Oracle GoldenGate Veridata Agent for the confirmation step. Do not use without the guidance of an Oracle support analyst.
`-rTs <trace_number>`	Turns on tracing for Oracle GoldenGate Veridata Server. <trace_number> is a bitmask of server execution trace flags. The higher the level, the more detailed the trace data. Do not use without the guidance of an Oracle support analyst.

`-pS <source_partition_name> |`
`-pSq <source_sql_predicate> |`
`-pSA1 <source_ascii_start_key> |`
`-pSA2 <source_ascii_end_key> |`
`-pSH1 <source_hex_start_key> |`
`-pSH2 <source_hex_end_key>`
	Runs the comparison using an existing source row partition or using an override partition that is defined by partition criteria. These options are mutually exclusive. They are valid only if comparing one compare pair (-j with -g and -c) and are ignored otherwise.
	-pS <source_partition_name>
	Specifies an existing source partition that is already defined and stored in the repository. The partition name is not validated and is passed directly to Veridata Server. There will be an error if the specified partition does not exist.
	-pSq <source_sql_predicate>
	Specifies a SQL predicate that defines a partition to override an existing source partition for a SQL table. The predicate is the conditional statement that follows the WHERE keyword, for example: LAST_NAME BETWEEN "A" AND "M". Do not include the WHERE keyword. It will be added automatically at runtime.
	If the predicate contains multiple words, it must be enclosed within quotes to make it a single command argument. The type of quote is dependent on the command shell or interpreter that is being used.
	If the predicate contains special characters (such as $, *, < in sh/csh or %, < in Windows), they must be properly escaped for that shell or interpreter.
	-pSA1 <source_ascii_start_key>
	Specifies an ASCII key as the starting key value of a partition that overrides an existing source partition for an Enscribe file.
	-pSA2 <source_ascii_end_key>
	Specifies an ASCII key as the ending key value of a partition that overrides an existing source partition for an Enscribe file.

Table 8 Vericom runtime arguments (continued)

Argument	Description
	-pSH1 <source_hex_start_key> Specifies a hexadecimal key as the starting key value of a partition that overrides an existing source partition for an Enscribe file. **-pSH2 <source_hex_end_key>** Specifies a hexadecimal key as the ending key value of a partition that overrides an existing source partition for an Enscribe file.

`-pT <target_partition_name> |`
`-pTq <target_sql_predicate> |`
`-pTA1 <target_ascii_start_key> |`
`-pTA2 <target_ascii_end_key> |`
`-pTH1 <target_hex_start_key> |`
`-pTH2 <target_hex_end_key>`
	These options specify target partitions and have the same rules as the corresponding options that specify source partitions.

`-pq <sql_predicate>`	Specifies a SQL predicate to be used for both the source and target SQL tables, as an override to existing partitions. This option has the same rules as -pSq <source_sql_predicate> and -pTq <target_sql_predicate>.
`-w | { -wp <poll_interval> }`	Directs vericom to wait for the job to finish before returning the prompt. Normally, vericom returns immediately after starting a job. ◆ -w waits the default time of one minute. ◆ -wp <poll_interval> waits the specified number of minutes and overrides the default. When either option is used, vericom polls the server until the job finishes, and then the statistics of the run are displayed. If multiple -w or -wp flags are specified, the last one on the command line takes effect.
`-rd0 | -rdN <run_ID>`	Valid for NonStop SQL/MP and Enscribe only. Controls delta processing for a compare pair. ◆ -rd0 disables delta processing for this run. All rows are compared. ◆ -rdN <run_ID> enables delta processing using a previous job run as the basis for the delta. For <run_ID> use the number from the Run ID line at the beginning of the job comparison report. Vericom does not validate the run ID that is supplied. To use these options, you must specify a compare pair with: ◆ -j ◆ -g ◆ -c

Table 8 Vericom runtime arguments (continued)

Argument	Description
`-{url | u} <url>`	Specifies the URL for Oracle GoldenGate Veridata Server in the form of <host>:<port>. Required if a run is not executed with veridata/server/bin as the working directory. If -url is not specified, vericom will try to determine the URL from the veridata.loc and veridata.cfg files.
`-{password | p} <password>`	Specifies the password or key of Oracle GoldenGate Veridata Server. Required if you do not execute a run with veridata/server/bin as the working directory. ◆ <password> must match the text that is in the veridata/shared/conf/password.txt file. This password is generated by the installer program. If -password is not specified, vericom will try to open password.txt to obtain the password.

Vericom exit statuses

Vericom exits with one of the following statuses.

Status	Description
0	The command executed successfully. If a job was run, it finished with all rows in-sync. If -w or -wp was not specified, the exit status is 0 if the job started successfully.
1	Invalid vericom syntax was used. For example, the following are invalid: vericom -helptun (Typographical error.) vericom -j -g group1 (The name of the job is missing.)
2	Vericom could not find connection information by looking in veridata.loc or veridata.cfg, and connection parameters were not specified with command arguments.
3	Provides more granularity for input errors that involve flags that run comparisons. For example, the following mistakes will cause this error: vericom –j job1 –c address=address In the preceding example, the -g <group> input is missing. It is required with -j if -c is used. vericom –j job1 –g group1 –rd0 In the preceding example, the -rd0 flag requires –c because delta processing applies at the compare pair level.
4	The job ran successfully, but there were rows that had a comparison status of something other than in-sync.
5	There was a communication error with Oracle GoldenGate Veridata Server.

Vericom output examples

To view the results of a comparison that is run with vericom, you can use Oracle GoldenGate Veridata Web to view the comparison report (see page 11), and you can view the output that is returned by vericom to the terminal. If a run finishes successfully, statistics for the job are displayed.

Example 1 The following example shows a run on a Windows system of TestJob without specifying -w. The process exits with status 0, and finished job statistics are not displayed.

```
C:\veridata\server\bin>vericom -j TestJob
Connecting to: localhost:9177
Run ID: (2256, 0, 0)
C:\veridata\server\bin> if errorlevel 0 echo EXITED 0 STATUS
EXITED 0 STATUS
```

Example 2 The following example shows a run of TestJob with-w specified. The process exits with status 4 because one of the compare pairs had a validation error. Finished job statistics are displayed.

```
C:\veridata\server\bin>vericom -j TestJob -w
Connecting to: localhost:9177
Run ID: (2257, 0, 0)
Job Start Time: 2008-03-21 22:48:05
Job Stop Time: 2008-03-21 22:48:20
Job Report Filename: C:\testjunit\rpt\TestJob\00002257\TestJob.rpt
Number of Compare Pairs: 3
Number of Compare Pairs With Errors: 1
Number of Compare Pairs With OOS: 1
Number of Compare Pairs With No OOS: 1
Number of Compare Pairs Cancelled: 0
Job Completion Status: WITH ERRORS
C:\veridata\server\bin> if errorlevel 4 echo EXITED 4 STATUS
EXITED 4 STATUS
```

Example 3 The following example shows a run of compare pair TABLE9=TABLE9 in job TestJob with -w specified. The process exits with status 0 because the tables are in-sync. Finished job statistics are displayed.

```
C:\veridata\server\bin>vericom -j TestJob -g TestGroup -c TABLE9=TABLE9 -w
Connecting to: localhost:9177
Run ID: (2258, 0, 0)
Job Start Time: 2008-03-21 22:51:08
Job Stop Time: 2008-03-21 22:51:11
Job Report Filename: C:\veridata\data\rpt\TestJob\00002258\TestJob.rpt
Number of Compare Pairs: 1
Number of Compare Pairs With Errors: 0
Number of Compare Pairs With OOS: 0
Number of Compare Pairs With No OOS: 1
Number of Compare Pairs Cancelled: 0
Compare Pair Report Filename:
C:\veridata\data\rpt\TestJob\00002258\TestGroup\CP_ TABLE9=TABLE9.rpt
Number of Rows Compared: 21
Number of Rows In Sync: 21
Number of Rows With Errors: 0
Number of Rows Out Of Sync: 0
Number of Inserts Out Of Sync: 0
Number of Deletes Out Of Sync: 0
Number of Updates Out Of Sync: 0
Compare Pair OOSXML Directory:
C:\veridata\data\oosxml\TestJob\00002258\TestGroup
Compare Pair OOSXML Filename:
Job Completion Status: IN SYNC
C:\veridata\server\bin> if errorlevel 0 echo EXITED 0 STATUS
EXITED 0 STATUS
```

On Unix systems, the exit status of the previously executed command is in the special variable '$?' if you are using SH or KSH shells. If you are using the CSH shell, the exit status of the previously executed command is in the special variable '$status'.

CHAPTER 11

Oracle GoldenGate Veridata Server configuration parameters

When using server-side sorting, you can use these parameters to adjust different aspects of the Nsort memory configuration. Nsort is the sorting library API that is built into Oracle GoldenGate Veridata server.

Overview of the server memory

Oracle GoldenGate Veridata Server uses virtual memory in the following ways:

- **Server memory for basic operation**. This is the amount of virtual memory that the server needs simply to operate. It stores object pools, database access libraries, and other information. This is usually about 30 MB.
- **Nsort memory**. This is the memory that is used by Nsort when server-side sorting is used. The virtual memory for Nsort is allocated for the entire comparison, not per thread. The rows are read from the agent and submitted to Nsort to be sorted. The Nsort sorting occurs in a thread that is separate from the thread that reads from the agent, and Nsort may use more threads to work in parallel. Once all of the rows from the agent are submitted to Nsort, the server process retrieves the sorted rows from Nsort for comparison.
- **Row hash queue memory**. This is the memory that buffers data between the agent processes, Nsort, and the server process. A comparison that uses database sorting requires a single queue each for the source and target. A comparison that uses server-side sorting requires two queues each for the source and target: one to buffer data from the agent, and one to buffer data from Nsort. Each queue has a capacity of 40 500k blocks. The memory usage by the queues is affected by the relative speed of the comparison and by the data coming from the agent. The relative speed between the two agents also affects the memory usage. A larger differential in speed increases the amount of memory that is used, because the queue needs to buffer the data.
- **MOOS queue memory.** This is the memory that holds potentially out-of-sync records between the initial comparison and confirmation steps of a comparison. The size of the MOOS queue is limited to 50K of records. Memory usage is also dependent on the width of each record.
- **SOAP messaging memory**. This is the memory that is used to serialize the SOAP messages between the server and the UI.
- **IPC buffer memory**. This is the memory that is used to exchange messages between the server and the agent.
- **Scratch runtime transient memory**. This is virtual memory space.

The amount of memory that can be used by Nsort cannot be greater than the minimum of:

- System physical memory
- Available memory in swap
- Current process limit (rlimit)

The Nsort library performs a check for available system virtual memory before allocating memory. The allocation can be affected by other processes that are running on the local system. The memory check is operating-system dependent, and not all checks are performed on all platforms.

When checking, Nsort assumes that it is the only Nsort in the current process. If the server employs multiple Nsort processes, however, this check can be inaccurate. This check also can be inaccurate if the free virtual memory has become fragmented over time. Even if the memory check succeeds, the actual allocation could fail if a contiguous chunk of memory is not available. This causes the comparison to fail. To overcome these uncertainties, use the server configuration parameters.

Estimating memory usage

Here is how Oracle GoldenGate Veridata Server uses memory resources on a per-comparison basis and among the overall workload.

Except for fixed virtual memory of 30 MB for basic tasks, Oracle GoldenGate Veridata Server uses the remaining virtual memory to perform comparisons.

To estimate the maximum number of concurrent comparisons

{Available virtual memory - 30 MB fixed memory} / {Average memory used per comparison}

Where: Average memory used per comparison assumes the following:

- Half of the comparisons will use server-side sorting (Nsort).
- The producer and consumer queues between the server and agents are fully utilized (500 Kb * 40 per queue). Two queues are needed for database sorting. Four queues are needed for server-side sorting.
- The maximum Nsort memory per comparison is set to the default.
- There is a fixed overhead for threads per comparison equal to 2Kb per thread stack plus two fixed. Database sorting requires four threads. Server-side sorting requires 14 threads.

In general this amount is about 80 MB of virtual memory per comparison.

To estimate the amount of memory used per row

((number of columns in key + 1) * 4) + 14 + size of key columns as ascii

For example, if a key is composed of one, 4-byte integer column, the memory required is:

((1 + 1) * 4) + 14 + 4 = 26 bytes of memory for this row

NOTE This assumes that non-key columns are compared by using a hash, not literally. More space is needed for literal comparisons.

How to set a parameter

To set a parameter, edit its entry in the veridata.cfg file. This file is stored in the shared/conf directory within the Oracle GoldenGate Veridata Server installation directory.

- Open an Oracle service request before changing these parameters. For more information, go to http://support.oracle.com.
- After changing a parameter value, stop and start Oracle GoldenGate Veridata Server.

Parameter descriptions

server.max_nsort_memory

Sets the maximum amount of Nsort virtual memory that is available to all running comparisons that use server-side sorting. The default amount assumes that half of the comparisons will use server-side sorting. You can reduce this value to reduce the impact of the server process on your system.

If a comparison cannot get enough virtual memory when it starts, it fails or tries again. If a comparison does get enough virtual memory, the currently available Nsort virtual memory gets decremented by the amount that the comparison reserves. When a comparison completes, it increments the amount of available Nsort virtual memory by the amount of Nsort virtual memory that it had reserved.

Syntax

```
server.max_nsort_memory {default | <n>{ M | m }}
```

- default allows Oracle GoldenGate Veridata to define a maximum value that is dependent on the operating system.
- <n>{ M | m } specifies a value in megabytes. For example, 1000M means a limit of 1000 megabytes.

server.max_concurrent_comparison_threads

Sets the maximum number of concurrent comparisons that can be executed. In general, the amount configured by the server is the optimal value, given the machines resources. You can lower this number to reduce the impact of the server on your system.

Syntax

```
server.max_concurrent_comparison_threads {default | <n>}
```

- default allows Oracle GoldenGate Veridata to compute the maximum number of concurrent threads based on average usage and available resources.
- <n> is a positive integer that sets the maximum number of concurrent comparison threads.

server.max_comparison_nsort_memory

Sets the maximum amount of virtual memory that can be reserved by a single comparison that is using Nsort. The default is 50 MB for the source data and 50 MB for the target data. A comparison that tries to reserve more than this amount will fail, even if it is configured to retry.

If the value of this parameter is changed, it might cause the concurrent comparisons that use server-side sorting to fail or retry, depending on the sorting parameters defined in the profile that is being used.

Syntax

```
server.max_comparison_nsort_memory {default | <n>{ M | m }}
```

- default specifies that there is no per-comparison limit.
- <n>{ M | m } specifies a value in megabytes. For example, 600M sets the limit for any comparison to be 600 megabytes.

server.nsort_waiting_threshold

Sets a threshold above which new jobs will fail if there are too many comparisons waiting to retry an Nsort virtual memory reservation.

Syntax

```
server.nsort_waiting_threshold {default | <n>}
```

- default allows all jobs to run, regardless of the number of comparisons that are in retry state.
- <n> sets the number of comparisons that can be waiting before new jobs will fail. For example, a value of 100 means that if there are 100 comparisons waiting to retry a memory reservation, and there is a job that is ready to begin execution, this job will not be allowed to run.

server.nsort_compare_function_pool_size

Sets the size of the Nsort compare function pool. When a comparison tries to reserve Nsort virtual memory, it will fail or retry if there are no free Nsort compare functions. If a comparison can reserve the required amount of Nsort virtual memory, it will use an Nsort compare function, thereby reducing the amount of those functions that are available. When the comparison completes, it will return its compare function to the pool.

Syntax

```
server.nsort_compare_function_pool_size {0 | <n>}
```

- 0 allows Oracle GoldenGate Veridata to define a default value that is dependent on the operating system.
- <n> is a positive integer that sets the compare function pool to the given size.

Additional parameters in the web interface

You can set these additional parameters within a profile in the Oracle GoldenGate Veridata Web interface.

Sorting Retry Limit

Controls the number of times that a comparison is retried if it cannot reserve the amount of virtual memory that it needs. The default of 0 prevents the comparison process from trying to reserve virtual memory if there is not enough available when it begins execution. A value of greater than 0 causes the process to try again to reserve virtual memory if its initial attempt fails.

The repeat attempt is executed after all of the other comparisons in the current job have been executed. If the profile option **Sorting retry interval** is greater than 0, the comparison process waits the specified interval before making a repeat attempt. Valid values are from 0 to 1000.

Sorting Retry Interval

Controls the amount of time that a comparison waits between attempts to reserve virtual memory if the initial attempt fails. Valid values are 0 to 999999999 seconds. A value of 0 means the comparison does not wait between attempts. A value of greater than 0 means the comparison waits for the specified interval before making another attempt. The default value is 600 seconds (10 minutes). Comparisons that fail to reserve virtual memory after the first attempt are retried after all of the other comparisons in that job have been executed.

CHAPTER 12
Upgrading Oracle GoldenGate Veridata

You must upgrade all components of Oracle GoldenGate Veridata. The use of different versions of the server and agent components is not supported.

Upgrading to a new version of the server and web components

Upgrade support overview

- **Upgrades from version 1.x to later versions**: There is no upgrade path from version 1.x to version 2.x or 3.x. You must recreate your configuration objects in the new version. You can install the new version, and then run the old and new versions side-by-side, so that you can refer to the existing configurations as you recreate them in the new version.
- **Upgrades from version 2.x using a new repository**: To change the database repository (to move from one instance of Oracle to another, for example, or to move from MySQL to SQL Server), follow the instructions in this manual to install the server and web components as a new installation. Oracle GoldenGate Veridata objects will be installed into the database that you specify during the installation, including the same tables for the user data that exist in the original repository. After the installation, you can use any migration tool to move the existing data from the old repository into the one.
- **Upgrades from version 2.x using the existing repository**: Use the following procedure, "Server and web upgrade instructions for all platforms". When installing a new version of Oracle GoldenGate Veridata Server and Oracle GoldenGate Veridata Web into an existing directory, the Oracle GoldenGate Veridata installer detects the existing installation and prompts you to upgrade or quit. The upgrade path retains the current Oracle GoldenGate Veridata database repository, without an option to drop any objects. Users can log on after the upgrade and continue to work with their existing environment.

Server and web upgrade instructions for all platforms

1. Start the Oracle GoldenGate Veridata repository database.
2. Stop all running comparisons or wait until they are finished.
3. Close all Oracle GoldenGate Veridata Web client sessions.
4. Shut down Oracle GoldenGate Veridata Server and Oracle GoldenGate Veridata Web. (For more information, see page 52.)

5. Run the Oracle GoldenGate Veridata installer:
   - UNIX and Linux: Run the GoldenGate_Veridata_<platform><version>.sh file from X-windows or an X-windows emulation program.
   - Windows: Run the GoldenGate_Veridata_<platform><version>.exe file.

   The following steps are the same on Windows, UNIX, and Linux. These instructions show the Windows version. Some prompts appear only in the Windows version.

6. **Welcome**: Click **Next** to start the installation.
7. **Select Destination Directory**: Select the Oracle GoldenGate Veridata installation that you want to upgrade.
8. Click **Upgrade** to upgrade the existing installation (and preserve the repository). If you do not want to upgrade this installation or preserve the repository, click **Quit**.
9. **Veridata Repository**: The installer displays the encrypted version of the last known password for the repository owner. Either accept the default or type the correct one.
10. **Veridata Server Configuration**: Accept the default to install the Microsoft Visual C++ Redistributable Package unless the system contains a version of this package that is at least as current as the one shown by the installer.
11. **Start after install**: Specify whether or not to start the software after the installation is finished. By default, it is installed to start manually. If you are installing it as a service, a system administrator can change the properties so that it starts automatically when the system starts.
12. **Information**: Review your selections and either click **Cancel** to start over or **Next** to start the upgrade.
13. To view the Oracle GoldenGate Veridata help system after the installation, accept the default and click **Finish**.

Upgrading to a new version of the agent

- As of Oracle GoldenGate Veridata version 3.0, you can use a Java agent for Oracle, SQL Server, and Teradata.
- To upgrade the agent on NonStop SQL/MP, you must use the version 3.0 C-agent.
- A version 3.0 C-agent is also available for the Oracle database as an upgrade.

Upgrading from a C-agent to a Java agent

There is no direct upgrade path from the C-agent to the Java-based agent. To use a Java agent:

- Follow the instructions in “Installing Oracle GoldenGate Veridata Java Agent” on page 21.
- You can specify the same port that was used by the existing agent Manager. If you use a different port number, make certain to notify the users of Oracle GoldenGate Veridata Web, so that they can update their connection configurations.

Upgrading a C-agent to a newer version

The instructions provided are for upgrading a C-agent from Veridata 2.x to a newer version. They assume that:

- You are upgrading the same agent on the same machine.
- You are keeping the existing file path names of the installation.
- If upgrading on a Windows system, you are keeping the existing Manager service name, if one exists.

To upgrade a C-agent on Windows, UNIX and Linux

1. Wait for all jobs that use this agent to finish running.
2. From the agent installation location, run GGSCI. The default location is GoldenGate_Veridata/agent.
3. In GGSCI, issue the following command to stop the Manager process.

   ```
   STOP MANAGER
   ```

   Shutting down the Manager process prevents new agent processes from being started by the server component, if it is still running.
4. Extract the new Oracle GoldenGate Veridata Agent files to the same directory where the current agent is installed. The default relative path that is extracted is veridata/agent. You might need to change the extraction path if your naming conventions are different. Overwrite the existing files in the agent directory with the extracted files.
5. Run GGSCI from the agent directory.
6. In GGSCI, issue the following command to start the Manager process.

   ```
   START MANAGER
   ```
7. In GGSCI, issue the following command to verify that the installation succeeded. The command should return “Manager is running.”

   ```
   INFO MANAGER
   ```
8. (Windows) In the Services control panel, verify that the correct service for this agent started.

To upgrade a C-agent on NonStop

1. Wait for all user jobs that use this agent to finish running.
2. From the agent installation location, run GGSCI. The default location is GoldenGate_Veridata/agent.
3. In GGSCI, issue the following command to stop the Manager process.

   ```
   STOP MANAGER
   ```

   Shutting down the Manager process prevents new agent processes from being started by the server component, if it is still running.
4. Transfer the new Oracle GoldenGate Veridata Agent files in binary mode to the volume and subvolume on the NonStop Server where you want to upgrade the agent.

5. Alter the VERUNPAK macro to be an edit file by issuing the following TACL command.

   ```
   FUP ALTER VERUNPAK, CODE 101
   ```

6. Run the VERUNPAK macro by issuing the following TACL command.

   ```
   RUN VERUNPAK
   ```

7. At the prompt, verify the location of the agent that you want to upgrade. Type `Y` to confirm the location shown or `N` to select another location.

   ```
   Installing GoldenGate at $DATA.GoldenGate Veridata
   Is this correct? (Y/N) y
   UNPAK - File decompression program - T1255G06 - (2002-05-06)
   Archive version: 1
   File Mode RESTORE Program - T9074G07 (15JAN2002)
   Copyright Tandem Computers Incorporated 1981-2002
   Summary Information
   Files restored = 7 Files not restored = 0
   GoldenGate Veridata for Nonstop Installation
   Installs the GoldenGate Veridata Product
   Enter X at any prompt to quit.
   ```

8. You are prompted for a SQL catalog for the agent to use. Type the catalog name or type `X` for no catalog.

   ```
   SQL Catalog for Compilation (X for no catalog)? $data.cpscat
   SQL compiling VERIAGT
   GoldenGate Veridata Installation Complete.
   ```

9. Run GGSCI from the agent directory.
10. In GGSCI, issue the following command to start the Manager process.

    ```
    START MANAGER
    ```

11. In GGSCI, issue the following command to verify that the installation succeeded. The command should return “Manager is running.”

    ```
    INFO MANAGER
    ```

12. If tables have remote partitions, copy VSNSERV to the remote nodes.

Copying VSNSERV to remote nodes

If your tables have partitions on remote nodes, you will need to place a copy of the VSNSERV module on each of those nodes.

If all of the remote nodes are the same hardware type, you can use a copy of the VSNSERV that is in the Oracle GoldenGate Veridata agent subvolume. Otherwise, you might need to download the correct agent build for that hardware type. It will include the correct VSNSERV.

To place the VSNSERV on each node, you can do either of the following:

- Install the entire Oracle GoldenGate Veridata Agent package on each of the remote nodes, even though the agent itself will not be running on them.
- Copy the VSNSERV object to each of the remote nodes. To use this option, take the following steps.

To copy VSNSERV to remote nodes

1. Copy the appropriate VSNSERV program to each of the remote nodes.
2. Log onto each remote node as a super user.
3. Issue the following commands on each remote node:

   ```
   FUP GIVE vsnserv, SUPER.SUPER
   FUP secure vsnserv, “NNNN”, PROGID
   ```

   - The first command sets the VSNSERV owner as `SUPER.SUPER`.
   - The second command sets security and PROGID to run as `SUPER.SUPER`.

4. This upgrade procedure assumes that the location of VSNSERV on each remote node did not change from the original installation. If the location did change, you must change the HOST parameter(s) in the GLOBALS file. See “Creating a GLOBALS file” on page 32.

CHAPTER 13
Uninstalling Oracle GoldenGate Veridata

Uninstalling the server and web components (all platforms)

1. Log on to the system as a user who has privileges to uninstall files.
2. Stop all running comparisons, or wait until they are finished.
3. Stop all Oracle GoldenGate Veridata Web client sessions.
4. Stop Oracle GoldenGate Veridata Server and Oracle GoldenGate Veridata Web. See page 52.
5. If you intend to drop the Oracle GoldenGate Veridata database repository, start the repository database.
6. Run the uninstall program from the Oracle GoldenGate Veridata directory:
   - *UNIX and Linux*: Using an X-windows or an X-windows emulation program, run uninstall.sh from the Oracle GoldenGate Veridata installation directory.
   - *Windows*: From the **Start** menu, go to the Oracle GoldenGate Veridata program folder, and select **Oracle GoldenGate Veridata Uninstaller**. Alternatively, you can run the uninstall.exe executable file from the installation directory.

   The uninstaller operates in the same way on Windows, UNIX, and Linux.
7. By default, the Oracle GoldenGate Veridata repository is not removed, so that user data is preserved. To drop the Oracle GoldenGate Veridata repository, select:
   - (Oracle) **Yes, drop the user/schema**. Supply the login of the database user or schema that owns the repository.
   - (Other databases) **Yes, drop the database** and **Yes, drop the user** or **Yes, drop the login** (whichever applies to the database type). Supply authentication credentials that have privileges to drop database objects. In the case of Microsoft SQL Server, you can use integrated login if the datasource configuration accepts it.
8. Click **Next** to confirm and start the uninstallation.
9. Click **Finish** to exit the uninstaller program.
10. (Optional) Remove the Oracle GoldenGate Veridata installation directory. The uninstall program does not remove files in this directory that were created by other programs. Logs are an example of such files.

Uninstalling the Java agent from a Windows system

1. Log on to the system as a user who has privileges to uninstall files.
2. Stop all running comparisons, or wait until they are finished.
3. Stop the agent process (see page 52).
4. From the **Start** menu, go to the Oracle GoldenGate Veridata Agent program folder, and then select **Oracle GoldenGate Veridata Agent Uninstaller**. Alternatively, you can run the uninstall_veridata_agent.exe executable file from the installation directory.
5. Click **Next** on the welcome screen to start the removal process.
6. One of two messages is displayed, depending on whether the agent is still running. A running agent could indicate that server processes still are using this agent. Click **Next** to continue with the uninstallation process or click **Quit** to cancel it. If you continue with the uninstallation, the status is displayed.
7. Click **Finish** to exit the uninstaller program.
8. (Optional) Remove the Oracle GoldenGate VeridataAgent installation directory. The uninstall program does not remove files in this directory that were created by other programs. Logs are an example of such files.

Uninstalling the Java agent from a UNIX or Linux system

1. Log on to the system as a user who has privileges to uninstall files.
2. Stop all running comparisons, or wait until they are finished.
3. Stop the agent process (see page 52).
4. Remove the files in the agent installation directory.

Uninstalling the C-agent (all platforms)

1. Log on to the system as a user who has privileges to uninstall files.
2. Stop all running comparisons, or wait until they are finished.
3. From the agent installation location, run GGSCI. The default location is GoldenGate_Veridata/agent.
4. In GGSCI, issue the following command to stop the Manager process.

   ```
   STOP MANAGER
   ```

   Shutting down the Manager process prevents new agent processes from being started by the server component, if it is still running.

Optional:

5. Remove the files that are in the installation location.
6. (NonStop) Purge any copies of the file named $*.*.VSNSERV.

APPENDIX 1

Changing Oracle GoldenGate Veridata Web port numbers

After the Oracle GoldenGate Veridata server and web components are installed, you can change certain port numbers if they conflict with other programs.

Ports are specified in the configuration file of the Tomcat web server application that was installed with Oracle GoldenGate Veridata. The name of this file is as follows (example shows UNIX file structure):

<GoldenGate_Veridata_install_directory>/Oracle GoldenGate Veridata/server/web/conf/server.xml

- The first line in the file is `<Server port="8820" shutdown="SHUTDOWN" debug="0">`. This is the port that the web server listens on for a shutdown message. You can change this port as needed.
- About halfway into the script, there are the following lines that specify ports:

```
<Service name="Tomcat-Standalone">
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8830" minProcessors="5" maxProcessors="75"
enableLookups="true" redirectPort="8443"
acceptCount="100" debug="0" connectionTimeout="20000"
useURIValidationHack="false" disableUploadTimeout="true" />
```

  - The first port number is the http port that users type on the address line when starting Oracle GoldenGate Veridata Web from a browser. You can change this port.
  - The second port is used internally. Do not change this port unless it conflicts with another program.

APPENDIX 2

Configuring an ODBC connection for a SQL Server repository

To use a SQL Server database as a repository, you must establish a system data source name (DSN). A DSN stores information about how to connect to the database.

NOTE You can use these same steps to edit or verify an existing DSN.

1. Run one of the following ODBC clients:
   - ❍ If using a 32-bit version of Oracle GoldenGate Veridata on a 64-bit system, create the DSN by running the ODBCAD32.EXE client from the %SystemRoot%\Windows\SysWOW64 folder.
   - ❍ If using a 64-bit version of Oracle GoldenGate Veridata 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 Veridata 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.
3. Do one of the following:
   - ❍ To create a new DSN, click **Add**.
   - ❍ To verify an existing DSN, select the name from the **System Data Sources** list and then click **Configure**.
4. (New data source only) Under **Create New Data Source**, select either the **SQL Server** or **SQL Native Client** driver.
5. Click **Finish**. A new wizard is displayed.

6. Supply or verify the following:
   - **Name**: Can be of your choosing.
   - **Description**: (Optional) Type a description of this data source.
   - **Which SQL Server do you want to connect to**: Select (or type) the name of the SQL Server instance. To create the DSN for a named instance, specify the host name and instance name in this format: <host>\<instance>. To create a DSN for the default instance, just the host name is sufficient. If the instance is on the local machine, you can select the localhost option.
7. Click **Next**.
8. Click the **Client Configuration** button.
9. In the **Edit Network Library Configuration** popup:
   - Enter a server alias name (optional).
   - Deselect the **Dynamically determine port** box, and then enter the name of a port number to use (the default of 1433 is acceptable). Oracle GoldenGate Veridata requires the use of a static port.
   - Under **Network Libraries**, make certain that TCP/IP is selected (default).
10. Click **OK** .
11. Based on your SQL Server configuration, select either **With Windows NT authentication** to require a network login or select **With SQL Server authentication using a login** to require database credentials. If you select SQL Server authentication, supply a login and password to test the DSN. This is for test purposes, only. This is not necessarily the user that will be used by Oracle GoldenGate Veridata.
12. Click **Next**.
13. You can leave all of the **Microsoft SQL Server DSN Configuration** settings to their default values.
14. Click **Next**.
15. You can leave all of the **Create New Data Source to SQL Server** settings to their default values.
16. Click **Finish**.
17. Click **Test Data Source** to test the connection.
18. Close the confirmation box and the **Create a New Data Source** box.

APPENDIX 3

Glossary of Oracle GoldenGate Veridata Terms

The following explains terminology contained in this manual.

Term	Definition
agent	See Oracle GoldenGate Veridata Agent.
client	An interface to Oracle GoldenGate Veridata Server, from which you configure or execute comparison work. Oracle GoldenGate Veridata provides a GUI client named Oracle GoldenGate Veridata Web and a command client named vericom, which provides a subset of the functions that can be performed from the web interface.
column mapping	An Oracle GoldenGate Veridata Web configuration object that establishes the correct structural relationship between the source and target objects in a compare pair.
compare pair	A configuration object in Oracle GoldenGate Veridata Web that associates one source table or file with one target table or file for the purpose of comparing their data. The source table typically contains data that is known or presumed to be up-to-date and accurate. The target table is a different table which contains identical data, such as a backup, that is synchronized with that of the source by some mechanism, such as replication.
comparison format	A standardized format to which Oracle GoldenGate Veridata transforms the native data type of a column. The use of standardized formats enables Oracle GoldenGate Veridata to support heteogeneous comparisons (those between different types of databases) where source and target data types are different, but similar.
comparison report	A report that is created at the conclusion of a comparison that contains information about the objects that were compared, the number of rows that are out-of-sync, and statistics about performance.
confirmation step, or confirm-out-of-sync (COOS) step	A follow-up comparison that is performed after the initial comparison step to confirm a row's status as in-sync or perpetually out of synchronization. The confirmation step waits until after a specified replication latency threshold has expired, to give a replication mechanism time to post any change that was in-flight.

Term	Definition
connection	A configuration object in Oracle GoldenGate Veridata Web that stores information about how the Oracle GoldenGate Veridata Server process will connect to a Veridata Agent process. It consists of a DNS host name or IP address, a Manager port number, and a datasource type.
datasource	A set of stored data, such as a relational or hierarchical database.
delta processing	A performance feature that Oracle GoldenGate Veridata supports on the NonStop platform by which Oracle GoldenGate Veridata detects which data blocks in the database were modified since a previous comparison and only compares the rows in those blocks. Rows in unchanged blocks are skipped.
digital signature	A message signed with a private key that can be verified by a person or process that has the public key, thereby proving that the message is unchanged and authentic.
GGSCI	The command line interface to the Oracle GoldenGate replication software, which also is used as the command interface to the Manager process of the Oracle GoldenGate Veridata Agent.
GLOBALS file	A parameter file that is required on some platforms where Oracle GoldenGate Veridata Agent is installed. It stores parameters that may be required in some installation conditions.
group, or compare group	A configuration object in Oracle GoldenGate Veridata Web that is a logical container for one or more compare pairs (sets of related source and target tables) and their associated datasource connections.
hash	A method of converting data into a unique numeric value (hash value) by substituting or transposing the data. If two hash values are different, then the degree of certainty that the two original input values are identical is extremely high (but not absolute). To be absolutely certain that values are identical, you can use a literal comparison.
in-flight	One of the possible states of a row during a comparison. The target row was out-of-sync in the initial comparison step, but was since updated and still do not match those of the source.
in-sync	One of the possible states of a row during comparison. The target row values match the source values that were retrieved during the initial comparison step.
initial comparison step, or row-hash step	The first step in a comparison, in which rows are retrieved from the source and target tables with a query and then compared. Target rows that appear to be out-of-sync are stored in a maybe out-of-sync (MOOS) queue in memory. The rows will be assessed further by the confirmation step.

Term	Definition
install	A program that is used on Windows platforms to install Oracle GoldenGate Veridata programs as a Windows service.
job	A unit of work that is linked to one or more compare groups that are processed together.
key, or key columns	A unique identifier that Oracle GoldenGate Veridata uses to order rows for comparison. This could be a primary key, a unique index, or a user-defined identifier.
latency	The difference in time between when a change is made to source data and when that change is reflected in the target data.
literal comparison	A comparison that is performed value for value, as opposed to a comparison that uses a hash.
Manager	Coordinator program of the Oracle GoldenGate Veridata Agent component. It coordinates requests sent to the agent process by the Oracle GoldenGate Veridata Server process.
Maybe-out-of-sync queue, or MOOS queue	A section of memory that stores rows that were determined to be out-of-sync in the initial comparison step, until they can be confirmed as in-sync or out-of-sync in the confirmation step.
Oracle GoldenGate Veridata Agent	Runs on the system where a source or target database resides. It fetches and returns blocks of rows, determines whether rows are out-of-sync, returns column-level detail, and executes requests on the database. The agent is started at the request of the Oracle GoldenGate Veridata Server and is controlled by the Oracle GoldenGate Veridata Manager process.
Oracle GoldenGate Veridata Server	The comparison engine that coordinates Oracle GoldenGate Veridata tasks, compares data, confirms out-of-sync data, and produces reports. Row sorting also can be performed by the server if the server-side sorting option was purchased.
out-of-sync	The data values in a source row are not identical to those of the corresponding target row.
out-of-sync report	A report that is created at the conclusion of a comparison that provides summary and column-level details about out-of-sync rows and the operations that are needed to bring them back into synchronization.
partition	A selection criteria that is defined within Oracle GoldenGate Veridata Web or the vericom interface, which enables comparisons to be performed against a subset of the rows in a table or file.

Term	Definition
persistently out-of-sync	One of the possible states of a row during comparison. During the confirmation step, the target values have not changed since the initial comparison step and still are not the same as those of the source.
profile	A configuration object in Oracle GoldenGate Veridata Web that is a stored set of global processing parameters, each containing unique settings for a specific purpose.
replication	The process of duplicating sets of data to one or more other locations in near real-time for such purposes as recovery or decision support.
repository	A collection of database objects that persists configuration information created by Oracle GoldenGate Veridata Web users to disk, saving it permanently as a user environment.
row partition	See partition.
row-hash step	See initial comparison step.
server	See Oracle GoldenGate Veridata Server.
server-side sorting	The sorting of source and target rows for comparison when performed by Oracle GoldenGate Veridata Server.
source	The database or location that is trusted to contain the most accurate and current version of data in an environment that contains replica versions of that data.
target	A database or location that contains a secondary or replica set of data that corresponds to a primary, or source, set of data.
validation	Performed by Oracle GoldenGate Veridata to test source and target table structures for compatibility.
vericom	The commandline interface to Oracle GoldenGate Veridata.

Index