Oracle® GoldenGate

Installing and Configuring Oracle GoldenGate Veridata 12c (12.2.1.4)

F16424-17
August 2024

Oracle GoldenGate Installing and Configuring Oracle GoldenGate Veridata 12c (12.2.1.4),

F16424-17

Contents

Preface

Audience vii
Documentation Accessibility vii
Related Information vii
Conventions vii

1 Installation and Configuration Overview

1.1 Understanding Oracle GoldenGate Veridata Components 1-1
1.2 Understanding Oracle GoldenGate Veridata Installation Topology 1-1
1.3 Installation Roadmap for Oracle GoldenGate Veridata 1-3
1.4 Understanding Oracle GoldenGate Veridata Distribution 1-4

2 Oracle GoldenGate Veridata Prerequisites and System Requirements

2.1 General Information 2-1
2.2 Prerequisites for Installing Oracle GoldenGate Veridata 2-1
2.2.1 Verifying Certification, System, and Interoparability Requirements 2-1
2.2.2 Selecting an Installation User 2-2
2.2.2.1 About User Permissions 2-2
2.2.2.2 About Non-Default User Permissions on UNIX Operating Systems 2-2
2.2.2.3 Verifying that the Installation User has Administrator Privileges on Windows Operating Systems 2-3
2.2.3 About the Oracle Home Directory 2-4
2.2.4 About JDK Requirements for an Oracle Fusion Middleware Installation 2-4
2.2.5 About Database Requirements for an Oracle Fusion Middleware Installation 2-5
2.3 Supported Databases for Compare 2-6
2.4 Supported/Unsupported Databases for Repair 2-6
2.5 Supported/Unsupported Databases for Download Repair SQL 2-7
2.6 Oracle GoldenGate Veridata Agent System Requirements 2-7
2.6.1 Oracle GoldenGate Veridata Java Agent 2-7
2.6.1.1 Environment Requirements to Use a Java Agent 2-8
2.6.2 Oracle GoldenGate Veridata C-Agent 2-8
2.6.3 Disk and Memory Requirements for the Agent Component 2-9

2.6.4 Database Privileges for the Agent Component 2-9
2.7 Oracle GoldenGate Veridata Server System Requirements 2-10
2.7.1 Location for the Server Component 2-11
2.7.2 Disk and Memory Requirements for the Server Component 2-11
2.7.3 Repository Requirements for the Server Component 2-13
2.7.3.1 Required Database Privileges for Oracle GoldenGate Veridata Server 2-13
2.7.3.2 Required Environment Components for Oracle GoldenGate Veridata Server 2-13
2.8 Oracle GoldenGate Veridata Web User Interface Requirements 2-14
2.8.1 Location for the Web Component 2-14
2.8.2 Java Environment for the Web User Interface 2-14
2.8.3 Supported Browsers for the Web User Interface 2-14
2.8.4 Recommended Screen Resolution for the Web User Interface 2-14
2.8.5 Port Numbers for the Web User Interface 2-14
2.8.6 Security for the Web User Interface 2-14
2.9 SQL/MP Data Exclusions 2-14

3 Installing Oracle GoldenGate Veridata

3.1 Starting the Installation Program 3-1
3.2 Navigating the Installation Screens to Install Oracle GoldenGate Veridata 3-1

4 Creating the Oracle GoldenGate Veridata Repository Schema

4.1 Configuring a Certified Database 4-1
4.2 Starting Repository Creation Utility 4-1
4.3 Navigating the RCU Screens to Create the Schema 4-2

5 Configuring Domain for Oracle GoldenGate Veridata

5.1 Creating the Database Schema 5-1
5.2 Configuring a Domain 5-1
5.2.1 Starting the Configuration Wizard 5-1
5.2.2 Navigating the Configuration Wizard Screens to Configure the Domain 5-1
5.2.3 Configuring the WebLogic Server Domain in a Silent Mode 5-9
5.2.4 Updating the Oracle WebLogic Server Credentials 5-11
5.3 Performing Next Steps 5-11

6 Deploying and Configuring Oracle GoldenGate Veridata Agent

6.1 About Oracle GoldenGate Veridata Agent Deployment Scripts 6-1
6.2 Configuring Oracle GoldenGate Veridata Agent 6-2
6.3 Starting the Oracle GoldenGate Veridata Agent 6-2

6.4 Using the Oracle GoldenGate Veridata Agent Deployment Script in Debug Mode 6-2
6.5 Reload the Logging Properties of the Veridata Agent 6-3

7 Running Oracle GoldenGate Veridata as a Windows Service

7.1 Running Oracle GoldenGate Veridata as a Windows Service (for Server) 7-1
7.2 Running Oracle GoldenGate Veridata as a Windows Service (for Agent) 7-3

8 Configuring Oracle GoldenGate Veridata Agent Using Kerberos to Connect to Oracle Database

9 Configuring Oracle GoldenGate Veridata Agent Using Kerberos to Connect to Hive

10 Connecting Oracle GoldenGate Veridata to SSL-Enabled Oracle Database

11 Connecting Oracle GoldenGate Veridata to SSL-Enabled MySQL Database

12 Connecting Oracle GoldenGate Veridata to SSL-Enabled SQL Server Database

13 Connecting Oracle GoldenGate Veridata to SSL-Enabled PostgreSQL Database

14 Next Steps After Configuring an Oracle GoldenGate Veridata Domain

14.1 Verifying the Oracle GoldenGate Veridata Installation 14-1
14.1.1 Reviewing Installation Log Files 14-1
14.1.2 Verifying the Directory Structure 14-1
14.2 Starting the Servers 14-1
14.2.1 Starting the Administration Server 14-2
14.2.2 Starting the Managed Server using Veridata Scripts 14-3
14.3 Creating Oracle GoldenGate Oracle GoldenGate Veridata Users and Assigning Privileges 14-3

14.4 Launching Oracle GoldenGate Veridata Web User Interface 14-4

15 Deinstalling Oracle GoldenGate Veridata

15.1 Understanding Product Deinstallation 15-1
15.2 Stopping Oracle Fusion Middleware 15-1
15.2.1 Stopping Your Oracle GoldenGate Veridata Agent 15-1
15.2.2 Stopping the Oracle Fusion Middleware Environment 15-2
15.3 Removing Your Database Schemas 15-2
15.4 Deinstalling the Software 15-2
15.4.1 Starting the Deinstaller 15-2
15.4.2 Navigating the Deinstallation Screens 15-3
15.5 Removing the Oracle Home Directory Manually 15-3
15.6 Removing Your Domain and Application Data 15-3
15.7 Deinstalling Oracle Fusion Middleware Infrastructure 15-4
15.8 Reinstalling the Software 15-4

A Installing Oracle GoldenGate Veridata C-Agent

A.1 Installation Overview A-1
A.2 Installing the C-Agent on a NonStop System A-1
A.2.1 Installing the Oracle GoldenGate Veridata Agent Files A-1
A.2.2 Copying VSNSERV to Remote Nodes A-2
A.2.3 Creating a GLOBALS File A-3
A.2.4 Configuring Manager A-4

B Understanding the Oracle GoldenGate Veridata Directory Structure

B.1 Oracle Home Directory Structure B-1
B.2 Veridata Domain Home Directory Structure B-3

C Oracle GoldenGate Veridata Distribution

C.1 Downloading Oracle GoldenGate Veridata C-1
C.2 Downloading Oracle GoldenGate Veridata C-Agent and Java Agent for DB2 for i or DB2 z/OS C-1
C.3 Downloading the Product C-2

D Oracle GoldenGate Veridata Silent Installation

Preface

This document describes how to use the installation program to install and configure Oracle GoldenGate Veridata.

Audience

This document is intended for installers and system administrators who are installing, configuring, and running Oracle GoldenGate Veridata.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Accessible Access to Oracle Support

Oracle customers who have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Information

The other Oracle GoldenGate Veridata documents for this release are as follows:

- Upgrading Oracle GoldenGate Veridata
- Using Oracle GoldenGate Veridata
- Oracle GoldenGate Veridata Release Notes
- Installing Oracle GoldenGate Veridata

The Oracle GoldenGate Product Documentation Libraries are found at

https://docs.oracle.com/en/middleware/goldengate/index.html

For additional information on Oracle GoldenGate, refer to, https://www.oracle.com/middleware/technologies/goldengate.html

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, such as "From the File menu, select **Save**." Boldface also is used for terms defined in text or in the glossary.
`italic`, *italic*	Italic type indicates placeholder variables for which you supply particular values, such as in the parameter statement: `TABLE` *`table_name`*. Italic type also is used for book titles and emphasis.
`MONOSPACE, monospace`	Monospace type indicates code components such as user exits and scripts; the names of files and database objects; URL paths; and input and output text that appears on the screen. Uppercase monospace type is generally used to represent the names of Oracle GoldenGate parameters, commands, and user-configurable functions, as well as SQL commands and keywords.
UPPERCASE	Uppercase in the regular text font indicates the name of a utility unless the name is intended to be a specific case.
{ }	Braces within syntax enclose a set of options that are separated by pipe symbols, one of which must be selected, for example: {*`option1`* | *`option2`* | *`option3`*}.
[ ]	Brackets within syntax indicate an optional element. For example in this syntax, the `SAVE` clause is optional: `CLEANUP REPLICAT` *`group_name`* [`, SAVE` *`count`*]. Multiple options within an optional element are separated by a pipe symbol, for example: [*`option1`* | *`option2`*].

1

Installation and Configuration Overview

Oracle GoldenGate Veridata compares one set of data to another and identifies data that is out-of-sync. Follow the standard installation and configuration process for Oracle GoldenGate Veridata, described in this guide, to create the standard topology, which represents a sample starting topology for this product.
See **Step-by-Step Labs to Install and Configure Oracle GoldenGate Veridata**

To upgrade an existing Oracle GoldenGate Veridata installation, see Upgrading Oracle GoldenGate Veridata in Upgrading Oracle GoldenGate Veridata.

This chapter includes the following topics:

1.1 Understanding Oracle GoldenGate Veridata Components

The Oracle GoldenGate Veridata distribution installs the following components:

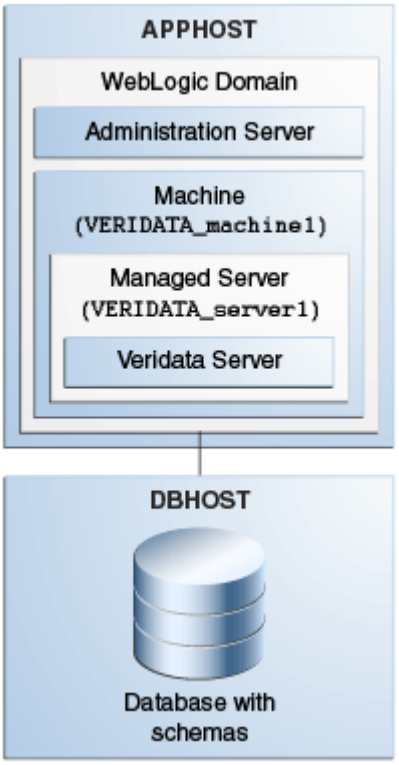
- Oracle GoldenGate Veridata Server
- Oracle GoldenGate Veridata Web User Interface
- Oracle GoldenGate Veridata Repository
- Oracle GoldenGate Veridata Agent
- Oracle GoldenGate Veridata Command-line Utilities

For more information about these components, see Starting and Stopping the Java-Based Components in Administering Oracle GoldenGate Veridata.

1.2 Understanding Oracle GoldenGate Veridata Installation Topology

Oracle GoldenGate Veridata Agent and Oracle GoldenGate Veridata Server are installed into an existing Oracle home containing Oracle Fusion Middleware Infrastructure.

Figure 1-1 Oracle GoldenGate Veridata Standalone Installation Topology



This topology represents a standard WebLogic Server domain that contains an Administration Server and a Managed Server on which the Infrastructure and Veridata Server are deployed. The domain is configured on a single host and requires a supported database where the Veridata Repository Database schema and required Fusion Middleware schema are installed.

All elements in this standard installation topology illustration are described in Table 1-1.

Table 1-1 Description of the Elements in the Standard Installation Topology

Element	Description and Links to Additional Documentation
APPHOST	Standard term used in Oracle documentation referring to the computer that is hosting the application tier.
DBHOST	Standard term used in Oracle documentation referring to the computer that is hosting the database.
WebLogic Domain	A logically related group of Java components (in this case, the administration Server, Managed Servers, and other related software components). For more information, see Understanding Key Concepts in the *Oracle Fusion Middleware Understanding Oracle Fusion Middleware Guide*.
Administration Server	The central control entity of a domain which maintains the domain's configuration objects and distributes configuration changes to Managed Servers. For more information, see What Is the Administration Server in the *Oracle Fusion Middleware Understanding Oracle Fusion Middleware Guide*.

Table 1-1 (Cont.) Description of the Elements in the Standard Installation Topology

Element	Description and Links to Additional Documentation
Machine	Logical representation of the computer that hosts one or more WebLogic Server instances (servers). Machines are also the logical glue between WebLogic Managed Servers and the Node Manager; in order to start or stop a Managed Server with Node Manager, the Managed Server must be associated with a machine.
Managed Server	Host for your applications, application components, Web services, and their associated resources. For more information, see Overview of Managed Servers and Managed Server Clusters in the *Oracle Fusion Middleware Understanding Oracle Fusion Middleware Guide*.
Infrastructure	Collection of services that include the following: • Metadata repository (MDS) This contains metadata for Oracle Fusion Middleware components, such as the Oracle Application Developer Framework. For more information, see What Is the Metadata Repository in the *Oracle Fusion Middleware Understanding Oracle Fusion Middleware Guide*. • Oracle Application Developer Framework (Oracle ADF) • Oracle Web Services Manager (OWSM) • Oracle Platform Security Services (OPSS)
Oracle GoldenGate Veridata Server	The Veridata Server coordinates the execution of Oracle GoldenGate Veridata tasks and processes information from multiple Veridata Agents.

1.3 Installation Roadmap for Oracle GoldenGate Veridata

This guide provides all the steps required to install and configure the standard Oracle GoldenGate Veridata topology. Within the procedures, the guide also provides references to additional information you can use if you want to create a slightly modified version of this topology.

Table 1-2 lists the steps required to install and configure the topology.

Table 1-2 Oracle GoldenGate Veridata Installation Roadmap

Task	Description	Documentation
Verify your system environment and install the prerequisites	Before beginning the installation, verify that the minimum system and network requirements are met.	See Oracle GoldenGate Veridata Prerequisites and System Requirements.
Obtain the appropriate distribution for your installation type	The Oracle GoldenGate Veridata distribution. For Oracle GoldenGate Veridata 12.2.1.4.0, the installer is `fmw_12.2.1.4.0_ogg.jar`.	See Oracle GoldenGate Veridata Distribution.
Determine your installation directories	Verify that the directories that will need to be created can be created or accessed by the installer, and exist on systems that meet the minimum requirements.	See "Understanding the Recommended Directory Structure" in *Planning an Installation of Oracle Fusion Middleware*.

Table 1-2 (Cont.) Oracle GoldenGate Veridata Installation Roadmap

Task	Description	Documentation
Install Oracle GoldenGate Veridata	Installing software transfers the software to your system and creates the Oracle home directory.	For more information about the install types available during installation, see Understanding Oracle GoldenGate Veridata Distribution. For installation instructions, see Installing Oracle GoldenGate Veridata.
Create Oracle GoldenGate Veridata Repository schema.	Use the Repository Creation Utility (RCU) to create the Repository database schemas.	See Creating the Oracle GoldenGate Veridata Repository Schema .
Create and configure your domain	Use the configuration wizard to create and configure your domain.	See Configuring Domain for Oracle GoldenGate Veridata to create the domain.
Apply patch using OPatch	Use OPatch to apply latest bundle patch, and execute repository patch if applicable.	See Next Steps After Configuring an Oracle GoldenGate Veridata Domain.

1.4 Understanding Oracle GoldenGate Veridata Distribution

The Oracle GoldenGate Veridata distribution is available as a `.jar` file. You must have a certified JDK already installed on your system in order to install and configure this distribution.

> **Tip:**
>
> For more information about distributions, see Obtaining Product Distributions in *Planning an Installation of Oracle Fusion Middleware*.

For information about how to obtain the distribution and also about the feature sets in the distribution, see Oracle GoldenGate Veridata Distribution.

2

Oracle GoldenGate Veridata Prerequisites and System Requirements

Before you begin the installation, find out the minimum system and network requirements and what pre-installation tasks you must first complete.
This chapter includes the following topics:

2.1 General Information

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

Oracle GoldenGate Veridata is designed to run on dedicated hardware to utilize all available CPUs and memory. Oracle recommends that you do not run any other application on the same host where the Oracle GoldenGate Veridata Server is installed.

2.2 Prerequisites for Installing Oracle GoldenGate Veridata

This section describes the prerequisites you must meet before you can successfully install and configure Oracle GoldenGate Veridata. The following sections explain the prerequisites:

- Verifying Certification, System, and Interoparability Requirements
- Selecting an Installation User
- About the Oracle Home Directory
- Verifying that the Installation User has Administrator Privileges on Windows Operating Systems

2.2.1 Verifying Certification, System, and Interoparability Requirements

Oracle recommends that you use the certification matrix and system requirements documents with each other to verify that your environment meets the requirements for installation.

1. **Verifying that your environment meets certification requirements:**

   Make sure that you install your product on a supported hardware and software configuration. See the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page.

   Oracle has tested and verified the performance of your product on all certified systems and environments. Whenever new certifications are released, they are added to the certification document right away. New certifications can be released at any time. Therefore, the certification documents are kept outside the documentation libraries and are available on Oracle Technology Network.

2. **Using the system requirements document to verify certification:**

   Oracle recommends that you use the *Oracle Fusion Middleware System Requirements and Specifications* document to verify that the certification requirements are met. For

example, if the certification document indicates that your product is certified for installation on 64-Bit Oracle Linux 6.5, use this document to verify that your system meets the required minimum specifications. These include disk space, available memory, specific platform packages and patches, and other operating system-specific requirements. System requirements can change in the future. Therefore, the system requirement documents are kept outside of the documentation libraries and are available on Oracle Technology Network.

3. **Verifying interoperability among multiple products:**

   To learn how to install and run multiple Fusion Middleware products from the same release or mixed releases with each other, see Oracle Fusion Middleware Interoperability and Compatibility in *Oracle Fusion Middleware Understanding Interoperability and Compatibility*.

2.2.2 Selecting an Installation User

The user who installs and configures your system must have the required permissions and privileges.

2.2.2.1 About User Permissions

The user who installs a Fusion Middleware product has the following permissions on them:

- Read and write permissions on all non-executable files (for example, `.jar`, `.properties`, or `.xml`). All other users in the same group as the file owner have read permissions only.
- Read, write, and execute permissions on all executable files (for example, `.exe`, `.sh`, or `.cmd`). All other users in the same group as the file owner have read and execute permissions only.

This means that someone other than the person who installs the software can use the installed binaries in the Oracle home to configure a domain or set of Fusion Middleware products.

Below are some additional considerations to make prior to running the installer:

- On UNIX operating systems, Oracle recommends that you set the `umask` to `027` on your system prior to installation. This ensures that file permissions are set properly during installation. Use the following command:

  ```
  umask 027
  ```

  You must enter this command in the same terminal window from which you plan to run the product installer.
- On UNIX operating systems, do not run the installation program as the `root` user. The installer startup validation will fail and you will not be able to continue.
- When managing a product installation (for example, applying patches), you must use the same user ID as was used to perform the initial product installation.
- On Windows operating systems, the user performing the installation must have Administrator privileges. See Verifying that the Installation User has Administrator Privileges on Windows Operating Systems.

2.2.2.2 About Non-Default User Permissions on UNIX Operating Systems

Changing the default permission setting reduces the security of the installation and your system. Oracle does not recommend you to change the default permission settings.

If other users require access to a particular file or executable, use the UNIX `sudo` command or other similar commands to change the file permissions.

Refer to your UNIX operating system Administrator's Guide or contact your operating system vendor, if you need further assistance.

2.2.2.3 Verifying that the Installation User has Administrator Privileges on Windows Operating Systems

By default, users with the administrator privilege sign in to the system with regular privileges, but can request elevated permissions to perform administrative tasks.

To perform a task with elevated privileges:

1. Find the Command Prompt icon, either from the Start menu or the Windows icon in the lower-left corner.
2. Right-click **Command Prompt** and select **Run as administrator**.

   This opens a new command prompt window, and all actions performed in this window are done with administrator privileges.

   > **Note:**
   >
   > If you have User Access Control enabled on your system, you may see an additional window asking you to confirm this action. Confirm and continue with this procedure.

   > **Note:**
   >
   > For Oracle Internet Directory, ensure that you have enabled User Account Control (UAC). If you have not done so already, enable UAC by following the instructions in the *Enabling User Account Control (UAC)* section from the appropriate version of *Oracle Fusion Middleware System Requirements and Specifications* for your installation.

   > **Note:**
   >
   > For Oracle Identity and Access Management components, ensure that you have enabled User Account Control (UAC). If you have not done so already, enable UAC by following the instructions in the *Enabling User Account Control (UAC)* section from the appropriate version of *Oracle Fusion Middleware System Requirements and Specifications* for your installation.

3. Perform the desired task.

   For example, to start the product installer:

   For a jar file, enter:

   ```
   java –jar distribution_name.jar
   ```

   For an executable (`.exe`, `.bin`, or `.sh` file), enter:

   ```
   distribution_name.exe
   ```

2.2.3 About the Oracle Home Directory

After installing the certified JDK for your operating system, use the Oracle Fusion Middleware 12.2.1.4.0 Infrastructure installer to install Oracle WebLogic Server 12.2.1.4.0 and the Oracle Java Required Files (JRF) Infrastructure services.

When you install any Oracle Fusion Middleware product, you must use an Oracle home directory.

This directory is a repository for common files that are used by multiple Fusion Middleware products installed on the same machine. These files ensure that Fusion Middleware operates correctly on your system. They facilitate checking of cross-product dependencies during installation. For this reason, you can consider the Oracle home directory a central support directory for all Oracle Fusion Middleware products installed on your system. Fusion Middleware documentation refers to the Oracle home directory as ORACLE_HOME.
Oracle Home Considerations

- Do not include spaces in the name of your Oracle home directory; the installer displays an error message if your Oracle home directory path contains spaces.
- You can install only one instance of each Oracle Fusion Middleware product in a single Oracle home directory. If you need to maintain separate versions of a product on the same machine, each version must be in its own Oracle home directory.

Although you can have several different products in a single Oracle home, only one version of each product can be in the Oracle home.
Multiple Home Directories

Although in most situations, a single Oracle home directory is sufficient, it is possible to create more than one Oracle home directory. For example, you need to maintain multiple Oracle home directories in the following situations:

- You prefer to maintain separate development and production environments, with a separate product stack for each. With two directories, you can update your development environment without modifying the production environment until you are ready to do so.
- You want to maintain two different versions of a Fusion Middleware product at the same time. For example, you want to install a new version of a product while keeping your existing version intact. In this case, you must install each product version in its own Oracle home directory.
- See Oracle Fusion Middleware Understanding Interoperability and Compatibility .

> **Note:**
>
> If you create more than one Oracle home directory, you must provide non-overlapping port ranges during the configuration phase for each product.

2.2.4 About JDK Requirements for an Oracle Fusion Middleware Installation

Most Fusion Middleware products are in `.jar` file format. These distributions do not include a JDK. To run a `.jar` distribution installer, you must have a certified JDK installed on your system.

Make sure that the JDK is installed *outside* of the Oracle home. If you install the JDK under the Oracle home, you may encounter problems when you try to perform tasks in the future. Oracle

Universal Installer validates that the Oracle home directory is empty; the install does not progress until you specify an empty directory. Oracle recommends that you locate your JDK installation in the `/home/oracle/products/jdk` directory.

Platform-specific distributions have a `.bin` (for UNIX operating systems) or `.exe` (for Windows operating systems) installer; in these cases, a platform-specific JDK is in the distribution and you do not need to install a JDK separately. However, you may need to upgrade this JDK to a more recent version, depending on the JDK versions that are certified.

Always verify the required JDK version by reviewing the certification information on the *Oracle Fusion Middleware Supported System Configurations* page. For 12c (12.2.1.4.0), the certified JDK is 1.8.0_221 and later.

To download the required JDK, navigate to the following URL and download the Java SE JDK:

http://www.oracle.com/technetwork/java/javase/downloads/index.html.

2.2.5 About Database Requirements for an Oracle Fusion Middleware Installation

Many Oracle Fusion Middleware products require database schemas prior to configuration. If you do not already have a database where you can install these schemas, you must install and configure a certified database.

> **Note:**
>
> Multi-tenancy feature is supported, that is, Pluggable Database (PDB) and Container Database (CDB) are supported.

To find a certified database for your operating system, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page on the Oracle Technology Network (OTN).

To make sure that your database is properly configured for schema creation, see *Repository Creation Utility Requirements* in the *Oracle Fusion Middleware System Requirements and Specifications* document.

After your database is properly configured, you use the Repository Creation Utility (RCU) to create product schemas in your database. This tool is available in the Oracle home for your Oracle Fusion Middleware product. See About the Repository Creation Utility in *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility*.

> **Note:**
>
> Based on your deployment topology and the work load, it is recommended that you refer to the following note on My Oracle Support, and take appropriate actions for your deployment.
>
> Performance Tuning Guidelines and Diagnostics Collection for Oracle Identity Manager (OIM) (Doc ID 1539554.1)

2.3 Supported Databases for Compare

Oracle GoldenGate Veridata supports the following databases for comparisons:

- Oracle, ADW, ATP, DBCS
- SQL Server
- MySQL, MariaDB (via MySQL JDBC connection string and driver)
- Teradata
- BigData Hive
- DB2 for i, DB2 LUW, and DB2 z/OS
- Informix
- Non Stop (Enscribe and SQL/MP)
- Sybase Adaptive Server Enterprise (ASE)
- PostgreSQL

Through its support website, Oracle provides late-breaking updates, discussions, and best practices about pre-upgrade requirements, upgrade processes, post-upgrade, compatibility, and interoperability. To find the latest information about Oracle GoldenGate Veridata release, including the list of certified database versions and operating systems, go to My Oracle Support at http://support.oracle.com.

2.4 Supported/Unsupported Databases for Repair

Oracle GoldenGate Veridata supports the following databases for repair functionality:

- Oracle, ADW, ATP, DBCS
- SQL Server
- Teradata
- DB2 for i, DB2 LUW, and DB2 z/OS
- Informix
- Sybase Adaptive Server Enterprise (ASE)
- MySQL, MariaDB (via MySQL JDBC connection string and driver)
- Non Stop (Enscribe, SQL/MP)
- PostgreSQL

Oracle GoldenGate Veridata doesn't support the following databases for repair functionality, but supported by Oracle GoldenGate:

- BigData Hive

Oracle provides late-breaking updates, discussions, and best practices about pre-upgrade requirements, upgrade processes, post-upgrade, compatibility, repair, and interoperability. For information about the latest release of Oracle GoldenGate Veridata, including the list of certified database versions and operating systems, go to My Oracle Support at http://support.oracle.com. For the support/certification matrix for the supported versions of databases at Oracle Support website or at http://www.oracle.com/technetwork/middleware/ias/downloads/fusion-certification-100350.html.

2.5 Supported/Unsupported Databases for Download Repair SQL

Oracle GoldenGate Veridata supports the following target databases for Download Repair SQL functionality:

- Oracle
- SQL Server

2.6 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.

Comparing Multi-Byte Data

The following considerations apply when you are comparing tables with multibyte data:

- A Java agent should be used for all platforms except NonStop, which has only a C-agent.
- The Java agent uses the UTF-8 character for comparing character data. Out-of-sync data is written to the report file using the UTF-8 character set.
- The Oracle GoldenGate Veridata Oracle C-Agent can be used for comparisons between Oracle databases where the source and target use the same character set and the host system uses the same byte order. An Oracle C-Agent cannot be used in a comparison with a Java agent. Character fields that contain characters that are not valid UTF-8 characters are displayed as hexadecimal in the out-of-sync reports.

2.6.1 Oracle GoldenGate Veridata Java Agent

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

- DB2 for i, DB2 LUW, and DB2 z/OS
- Oracle
- SQL Server
- Sybase ASE
- Teradata
- Informix
- BigData Hive

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.

2.6.1.1 Environment Requirements to Use a Java Agent

Windows, UNIX and Linux, All Supported Databases

A Java environment is required on all platforms. Download and install either the Java Developer Kit (JDK) or the Java Runtime Environment (JRE) from the Oracle website.

DB2

A TCP/IP port must be configured and active.

Oracle

The listener must be configured and running.

SQL Server

A static TCP/IP port must be configured and enabled.

Sybase

Know the ASE listen port.

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 get prompted for the location during the installation steps in this guide.

INFORMIX

A TCP/IP port must be configured and active.

Hive

A TCP/IP port must be configured and active.

ADW/ATP

For more information, see: Connecting Veridata Agent to ADW/ATP.

2.6.2 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 supported for the Oracle database, but as of release 11.2.1.0.0, it will no longer be enhanced.
- 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:

Environment Variables for Dynamically Linked C-Agents

Variable	Operating System
`PATH=Oracle_libraries`	Windows
`SHLIB_PATH=Oracle_libraries`	HP-UX
`LIBPATH=Oracle_libraries`	IBM AIX
`LD_LIBRARY_PATH=Oracle_libraries`	All other supported UNIX platforms

2.6.3 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.

2.6.4 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.

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`
- `EXECUTE_CATALOG_ROLE` (for `GET_TAG` and `SET_TAG` procedures)
- `CREATE TABLE` (for COOS Join in case of tables with no primary/unique constraint)
- `CREATE TABLESPACE` (for COOS Join in case of tables with no primary/unique constraint)

- `GRANT UPDATE, DELETE, INSERT` (for COOS Join in case of tables with no primary/unique constraint)

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 sub volumes 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.

Sybase ASE

- Access to the databases to be compared.
- `SELECT` privileges on the tables to be compared.
- `SELECT` privileges on the `sysdatabases` system table in the master database to view the list of databases available in the server.

Teradata

- `SELECT` privileges on the tables to be compared.

Informix

SELECT privileges on the tables to be compared.

BigData Hive

SELECT privileges on the tables to be compared.

Required Database Privileges for Using the Repair Feature

For all databases, the database user must have the `UPDATE`, `INSERT`, and `DELETE` privileges on the tables to be repaired.

For Sybase database, if the table has triggers and suppression of triggers enabled, the database user must have the `replication_role` privilege.

For SQL Server database, If the table being repaired has 'identity columns', the Repair User specified must be either the table owner, or should have `ALTER` permission on the table that is being repaired.

2.7 Oracle GoldenGate Veridata Server System Requirements

This section describes the installation location, additional programs, disk, memory, and repository requirements for Oracle GoldenGate Veridata Server.

2.7.1 Location for the Server Component

The server and web user interface 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 and web user interface components on a NonStop system. To use Oracle GoldenGate Veridata for NonStop databases:

- Install the server and web user interface 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.

2.7.2 Disk and Memory Requirements for the Server Component

The server component uses about 200 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 machine are the row sorting operations of the initial comparison step when using server-side sorting.

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 cols in key + 1) * 4) + 16 + (comparison width of a key col)
```

Where:

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

Comparison format data sizes:

Comparison Format	Data Size
Numbers	One byte for each significant digit. Leading zeros and trailing zeros after the decimal point (such as the right most zeros in 1234.500) are not counted.
Timestamp	19 to 32 bytes depending on the fractional precision.
Date	10 bytes.
Time	8 to 18 bytes depending on the fractional precision.
String	1 to 4 bytes per character for the UTF-8 encoding of the Java agent. The NonStop agent and the Oracle C agent use the database native character set.
Binary	The bytes as stored in the database.

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

`((1 + 1) * 4) + 16 + 9 = 33` 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

Oracle GoldenGate Veridata uses an external merge sort to sort the data. As data is received from the agent, the rows are sorted in memory. When a memory buffer is full, the sorted rows are written to disk.

In order to sort the data, the sort process matches the initial data set size for temporary storage space. The required amount of temporary space is determined by the number of rows, the row size, and the amount of available sort memory. The following cases illustrate the different modes of the sort depending on the available resources.

- **In-Memory Sort:** This sorts the data entirely in memory and is the fastest method, but the memory requirements may exceed what is available. The sort memory must be approximately 2.5 times larger than the size of the data set.
- **One Disk Pass:** This sorts data and writes to the disk only once. It requires sort disk space equal to the size of the data set. This process is almost as fast as the in-memory sort and the memory requirements are lower. In general the Oracle GoldenGate Veridata server can write the rows to disk faster than the agent can read them from the database.
- **Two Disk Passes:** This sorts and writes to the disk twice, requiring sort disk space twice the size of the data set. Although the disk requirement is greater, very large data sets can be sorted with a reasonable amount of memory.
- **Three or More Disk Passes:** After all of the rows have been received from the agent, additional sorting may be required before the rows are ready for the final write to disk. If it is necessary to access the disk three or more times, the required sort space will be three or more times the data set size. This is slow and should be avoided.

Beyond this allocation, memory is required for storing rows during the second step of processing, the confirmation step. This can be up to 20 MB 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.

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 the sort 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:

- The temporary space should be located on a reasonably fast file system. A network file system located on a remote server may slow the comparison processing.
- 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 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 Oracle GoldenGate Veridata Server Configuration Parameters in *Administering Oracle GoldenGate Veridata*.

2.7.3 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 interface.

Use the Repository Creation Utility (RCU) to create the repository. See Creating the Oracle GoldenGate Veridata Repository Schema .

You can use the following databases as a repository:

- Oracle
- SQL Server

2.7.3.1 Required Database Privileges for Oracle GoldenGate Veridata Server

Oracle

1. Create a user and password.
2. 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

2.7.3.2 Required Environment Components for Oracle GoldenGate Veridata Server

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. A login and password, if using an existing database user for the repository.

SQL Server

1. 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.

2. A login and password (or integrated authentication), if using an existing database user for the repository.

2.8 Oracle GoldenGate Veridata Web User Interface Requirements

This section describes the Java package, browsers, screen resolution, security, and ports recommended or required for the Oracle GoldenGate Web User Interface.

For more information, see Oracle Fusion Middleware System Requirements and Specifications.

2.8.1 Location for the Web Component

See Location for the Server Component. One installer program installs both Oracle GoldenGate Veridata Web Server component and Oracle GoldenGate Veridata Web User Interface.

2.8.2 Java Environment for the Web User Interface

JDK 1.8.0_221 or higher is required for the Veridata Web User Interface installation.

2.8.3 Supported Browsers for the Web User Interface

For more informtion, see Oracle Software Web Browser Support Policy.

2.8.4 Recommended Screen Resolution for the Web User Interface

Minimum of 1024 x 768

2.8.5 Port Numbers for the Web User Interface

- The Oracle GoldenGate Veridata Web application interacts with Oracle WebLogic Server. Oracle WebLogic Server is installed with Oracle GoldenGate Veridata Server on default ports. You can change the port numbers if needed.
- Oracle GoldenGate Veridata Web makes use of the default port 8830.

2.8.6 Security for the Web User Interface

The web server provides user security roles that control access to certain pages of the Oracle GoldenGate Veridata Web interface. To create users and define user roles, use the Oracle WebLogic Server Administration Console. For more information, see Securing Access to Oracle GoldenGate Veridata by Defining User Roles in *Administering Oracle GoldenGate Veridata*.

2.9 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.

3
Installing Oracle GoldenGate Veridata

This topic describes how to install a new, clean copy of the Oracle GoldenGate Veridata software. It gives you a clear view how to start the installation and how to navigate through the installation of the Oracle GoldenGate Veridata.

> **Note:**
>
> The Oracle GoldenGate Veridata agent has to remotely access database DB2 for iSeries and DB2 for z/OS. Oracle does not support installing the Oracle GoldenGate Veridata agent on these platforms.

This chapter includes the following topics:

3.1 Starting the Installation Program

To start the installation program for Oracle GoldenGate Veridata, perform the following steps.

1. Log in to the target system.
2. The installer requires that a certified JDK already exists on your system. For more information, see the appropriate certification document at *Oracle Fusion Middleware Supported System Configurations*.
3. Go to the directory where you downloaded the installation program. For more information about downloading the installer, see Oracle GoldenGate Veridata Distribution.
4. Launch the installation program by invoking `java -jar` from the JDK directory on your system, as shown in the example below:

   ```
   /home/Oracle/jdk8_221/jdk1.8.0_221/bin/java -jar fmw_12.2.1.4.0_ogg.jar
   ```

   Be sure to replace `/home/Oracle/jdk8_221/jdk1.8.0_221` in this example with the location of the JDK on your system.

When the installation program appears, you are ready to begin the installation. See Navigating the Installation Screens to Install Oracle GoldenGate Veridata for a description of each installation program screen.

3.2 Navigating the Installation Screens to Install Oracle GoldenGate Veridata

The installation program displays a series of screens, in the order listed in Table 3-1.

If you need additional help with any of the installation screens, click the screen name.

Table 3-1 Oracle GoldenGate Veridata Install Screens

Screen	Description
Installation Inventory Setup	This screen will appear if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location. For more information about the central inventory, see Understanding the Oracle Central Inventory in *Oracle Fusion Middleware Installing Software with the Oracle Universal Installer Guide*.
Welcome	This screen introduces you to the product installer.
Installation Location	Use this screen to specify the location of your Oracle home directory. For more information, see "Selecting Directories for Installation and Configuration" in *Planning an Installation of Oracle Fusion Middleware*.
Installation Type	Use this screen to select the type of installation and consequently, the products and feature sets you want to install. For Oracle GoldenGate Veridata the following are the installation types: • Oracle GoldenGate Veridata Server: Installs Veridata Web Server • Oracle GoldenGate Veridata Agent: Installs Veridata Java Agent • Oracle GoldenGate Veridata Server and Agent: Installs both Web Server and Java Agent
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements.
Security Updates	If you already have an Oracle Support account, use this screen to indicate how you would like to receive security updates. If you do not have one and are sure you want to skip this step, clear the check box and verify your selection in the follow-up dialog box.
Installation Summary	Use this screen to verify the installation options you selected. If you want to save these options to a response file, click **Save** and provide the location and name of the response file. Response files can be used later in a silent installation situation. For more information about silent or command line installation, see Using the Oracle Universal Installer in Silent Mode in *Installing Software with the Oracle Universal Installer* Guide.
Installation Progress	This screen allows you to see the progress of the installation.
Installation Complete	This screen appears when the installation is complete. Review the information on this screen, then click **Finish** to dismiss the installer.

4

Creating the Oracle GoldenGate Veridata Repository Schema

Learn how to use the Repository Creation Utility (RCU) to create the veridata database repository where your Oracle GoldenGate domain instances hosted.

This step is required before any domain is configured for your Oracle GoldenGate Veridata instance.

4.1 Configuring a Certified Database

Before you begin, make sure you have installed and configured a certified database. For more information about the supported database, see the certification matrix at `http://support.oracle.com`.

For more information, see Verifying that the Installation User has Administrator Privileges on Windows Operating Systems.

4.2 Starting Repository Creation Utility

Navigate to the *ORACLE_HOME*`/oracle_common/bin` directory and start RCU.

1. Set the *JAVA_HOME* environment variable to your JDK location.

   On UNIX operating systems:

   ```
   setenv JAVA_HOME/home/Oracle/Java/jdkversion
   ```

   On Windows operating systems, right-click on the desktop, then select **My Computer**, then select **Properties**. Click **Advanced System Settings** in the left column, then click **Environment Variables** in the System Properties window.

   In the Environment Variables window, set *JAVA_HOME* to your JDK location.

2. Navigate to the *ORACLE_HOME*`/oracle_common/bin` directory.

3. Start RCU.

   On UNIX operating systems:

   ```
   ./rcu
   ```

   On Windows operating systems:

   ```
   rcu.bat
   ```

Note:

For Oracle Autonomous Transaction Processing-Shared (ATP-S) database, Oracle GoldenGate Veridata supports only RCU silent mode for schema loading. Following is an example of an RCU command. However, you must ensure to modify the connection information according to your database details.

```
./rcu -silent -createRepository -connectString
'(description=(CONNECT_TIMEOUT=120)(RETRY_COUNT=20)(RETRY_DELAY=3)
(TRANSPORT_CONNECT_TIMEOUT=3)(ADDRESS_LIST=(LOAD_BALANCE=on)
(ADDRESS=(PROTOCOL=TCP)(HOST=host-qr7it-
scan.fleetsubnet.adbvcn.oraclevcn.com)(PORT=1521)))
(CONNECT_DATA=(SERVICE_NAME=FMWATPDEDIC_tp.atp.oraclecloud.com)))'
-dbUser admin -dbRole normal -useSSL -sslArgs
javax.net.ssl.trustStore=/home/opc/veridata/
cwallet.sso,javax.net.ssl.trustStoreType=SSO,javax.net.ssl.keyStore=/
home/opc/veridata/
cwallet.sso,javax.net.ssl.trustStoreType=SSO,oracle.net.ssl_server_dn_match=
true,oracle.net.ssl_version=1.2
-useSamePasswordForAllSchemaUsers true -selectDependentsForComponents  true
-schemaPrefix OGG18
-component UCSUMS -tablespace DATA  -temptablespace TEMP
-component OPSS -tablespace DATA -temptablespace  TEMP
-component IAU -tablespace DATA -temptablespace TEMP
-component STB -tablespace DATA -temptablespace TEMP
-component WLS -tablespace DATA  -temptablespace TEMP
-component IAU_APPEND -tablespace DATA  -temptablespace TEMP
-component IAU_VIEWER -tablespace DATA  -temptablespace TEMP
-component MDS -tablespace DATA -temptablespace  TEMP
-component VERIDATA -tablespace DATA -temptablespace TEMP
-serverDN "CN=adwc.uscom-east-1.oraclecloud.com,OU=Oracle BMCS
US,O=OracleCorporation,L=Redwood City,ST=California,C=US"
```

4.3 Navigating the RCU Screens to Create the Schema

Follow the instructions on this section to complete schema creation for Oracle GoldenGate Veridata.

Refer to Repository Creation Utility Screens in the *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide* for more information about any of the screens mentioned in this section.

Task 1 Introducing RCU
Click **Next**.

Task 2 Selecting a Repository Creation Method
If you have the necessary permission and privileges to perform DBA activities on your database, select **System Load and Product Load Concurrently**. This procedure assumes that you have the necessary privileges.
For Oracle database, if you do not have the necessary permission or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option will generate a SQL script, which can be provided to your database administrator. See About System Load and Product Load in the *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*.

Task 3 Providing Database Connection Details
Provide the database connection details for RCU to connect to your database.
Click **Next** to proceed, then click **OK** on the dialog window confirming that connection to the database was successful.

Task 4 Selecting Components
Select **Create new prefix**, specify a custom prefix, then select the Oracle GoldenGate Veridata Repository schema.
The custom prefix is used to logically group these schemas together for use in this domain only; you must create a unique set of schemas for each domain as schema sharing across domains is not supported.
For more information about custom prefixes, see About Custom Prefixes in the *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*.
For more information about how to organize your schemas in a multi-domain environment, see Planning Your Schema Creation in the *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*.

> **Tip:**
>
> You must make a note of the custom prefix you choose to enter here; you will need this later on during the domain creation process.

Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Task 5 Specifying Schema Passwords
Specify how you want to set the schema passwords on your database, then specify and confirm your passwords.

> **Tip:**
>
> You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.

Task 6 Map Tablespaces
Specify the Tablespaces for the Oracle GoldenGate Veridata Repository.

> **Tip:**
>
> More information about the fields on this screen can be found in *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*.

Task 7 Completing Schema Creation
Navigate through the remainder of the RCU screens to complete schema creation. When you reach the Completion Summary screen, click **Close** to dismiss RCU.

5

Configuring Domain for Oracle GoldenGate Veridata

Learn how to create and configure an Expanded WebLogic domain for veridata using the configuration wizard.
This chapter includes the following topics:

5.1 Creating the Database Schema

Before you can configure the topology for your agent, make sure you have created the necessary schema in your database. See Creating the Oracle GoldenGate Veridata Repository Schema for instructions.

5.2 Configuring a Domain

You can configure a WebLogic Server domain either using the Configuration wizard or in a silent mode.

For more information about other methods available for domain creation, see Additional Tools for Creating, Extending, and Managing WebLogic Domains in *Creating WebLogic Domains Using the Configuration Wizard*.

This topic contains the following:

- Starting the Configuration Wizard
- Navigating the Configuration Wizard Screens to Configure the Domain
- Configuring the WebLogic Server Domain in a Silent Mode

5.2.1 Starting the Configuration Wizard

To configure a domain, navigate to the *ORACLE_HOME*`/oracle_common/common/bin` directory and start the WebLogic Server Configuration Wizard.

On UNIX operating systems:

```
./config.sh
```

On Windows operating systems:

```
config.cmd
```

5.2.2 Navigating the Configuration Wizard Screens to Configure the Domain

Follow the instructions in this section to create and configure the domain for the topology.

> **Note:**
>
> You can use the same procedure described in this section to extend an existing domain. If your needs do not match the instructions given in the procedure, be sure to make your selections accordingly, or refer to the supporting documentation for additional details.

Domain creation and configuration includes the following tasks:

- Task 1, Selecting the Domain Type and Domain Home Location
- Task 2, Selecting the Configuration Templates
- Task 3, Configuring the Administrator Account
- Task 4, Specifying the Domain Mode and JDK
- Task 5, Specifying the Datasource Configuration Type
- Task 6, Specifying JDBC Component Schema Information
- Task 7, Testing the JDBC Connections
- Task 8, Specifying Keystore Details
- Task 9, Selecting Advanced Configuration
- Task 10, Configuring the Administration Server Listen Address
- Task 12, Configuring Managed Servers
- Task 13, Reviewing Your Configuration Specifications and Configuring the Domain
- Task 14, Reviewing Your Domain Home and Administration Server URL

Task 1 Selecting the Domain Type and Domain Home Location
On the Configuration Type screen, select **Create a new domain**.
In the Domain Location field, specify your Domain home directory.
It is recommended that you locate your Domain home in accordance with the directory structure summarized in "Understanding the Recommended Directory Structure" in *Planning an Installation of Oracle Fusion Middleware*, where the Domain home is located outside the Oracle home directory. This directory structure will help you avoid issues when you need to upgrade or reinstall your software.

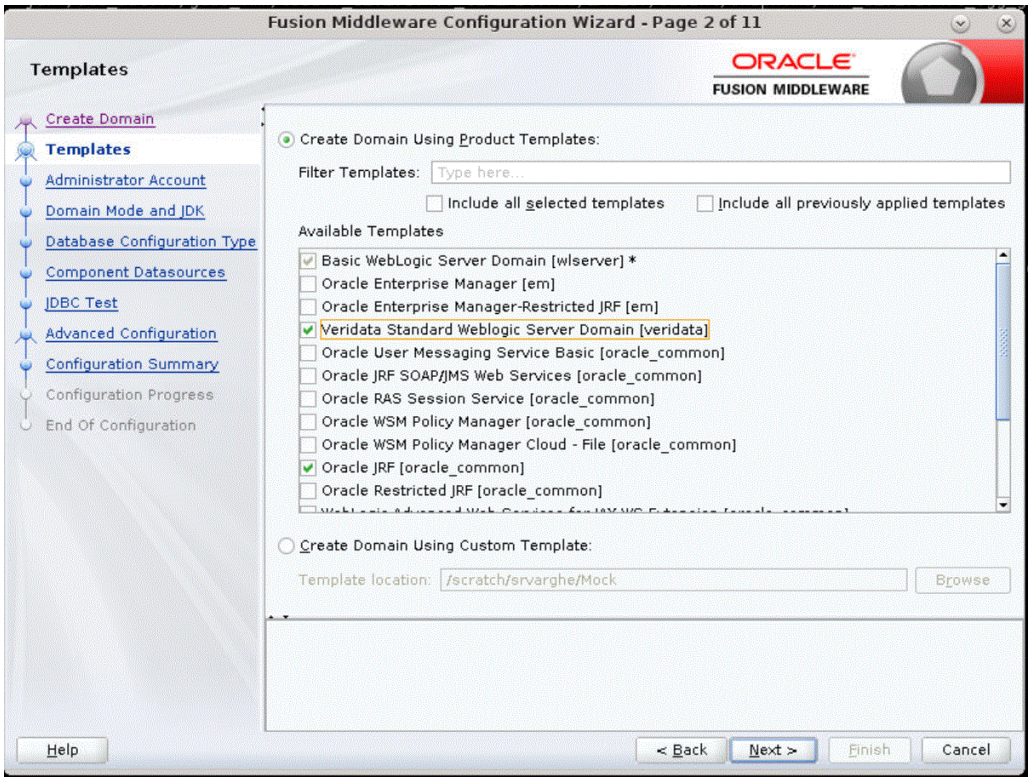
> **Tip:**
>
> More information about the other options on this screen can be found in Configuration Type in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*.

Task 2 Selecting the Configuration Templates
On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select **Oracle GoldenGate** from the Template Categories. By default, the following templates are selected:

- Veridata Standard WebLogic Server Domain [veridata]

Selecting this template automatically selects the following as dependencies:

- Oracle JRF [oracle_common]
- Basic WebLogic Server Domain [wlserver]



> **Tip:**
>
> More information about the options on this screen can be found in Creating WebLogic Domains Using the Configuration Wizard Templates in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*.

Task 3 Configuring the Administrator Account
On the Administrator Account screen, specify the user name and password for the default WebLogic Administrator account for the domain.
It is recommended that you make a note of the user name and password specified on this screen; you will need these credentials later to boot and connect to the domain's Administration Server.

Task 4 Specifying the Domain Mode and JDK
On the Domain Mode and JDK screen:

- Select **Production** in the Domain Mode field.
- Select the Oracle HotSpot JDK in the JDK field. JDK Version 1.8.0_101 or higher is required.

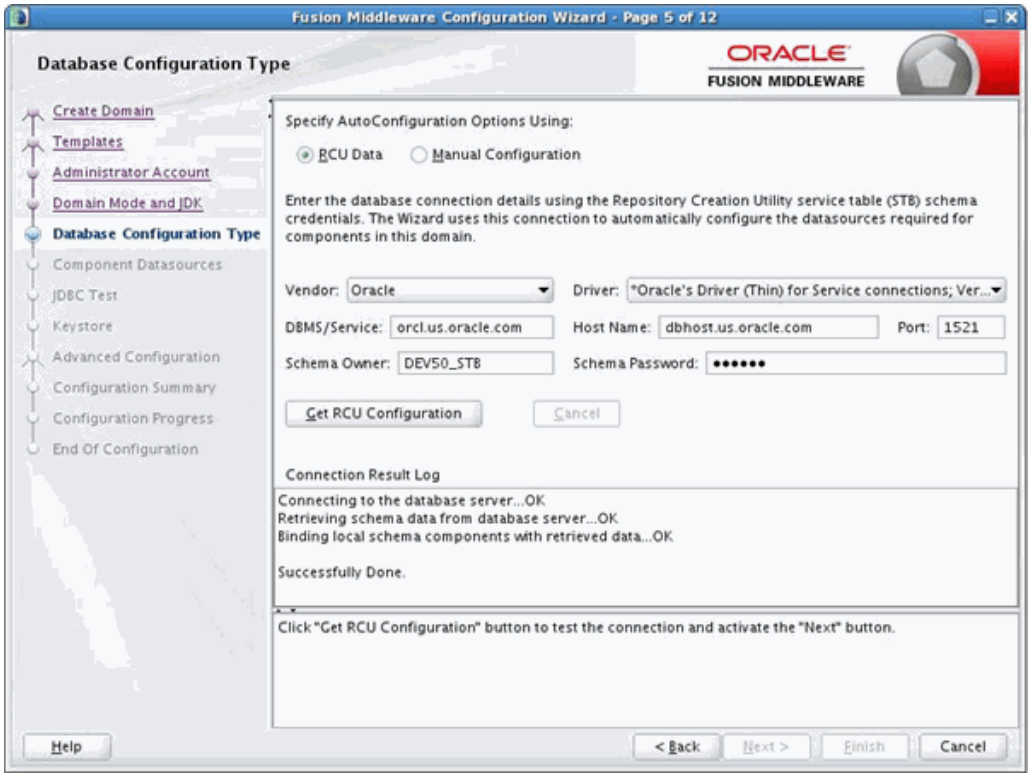
> **Tip:**
>
> More information about the options on this screen can be found in Domain Mode and JDK in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*.

Task 5 Specifying the Datasource Configuration Type
Select **RCU Data** to activate the fields on this screen. The **RCU Data** option instructs the Configuration Wizard to connect to the database and Service Table (STB) schema to automatically retrieve schema information for the schemas needed to configure the domain.

> **Note:**
>
> If you choose to select **Manual Configuration** on this screen, you will have to manually fill in the parameters for your schema on the JDBC Component Schema screen.

After selecting **RCU Data**, specify the database connection details in the following fields:

Field	Description
DBMS/Service	Enter the database DBMS name, or service name if you selected a service type driver.
Host Name	Enter the name of the server hosting the database.
Port	Enter the port number on which the database listens.
Schema Owner Schema Password	Enter the username and password for connecting to the database's Service Table schema. This is the schema username and password that was specified for the Service Table component on the "Schema Passwords" screen in RCU (see Creating the Database Schema). The default username is *prefix*_STB, where *prefix* is the custom prefix that you defined in RCU.



Click **Get RCU Configuration** when you are finished specifying the database connection information. A message is displayed indicating that the connection is successful.

> **Tip:**
>
> More information about the **RCU Data** option can be found in About the Service Table Schema in the *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*.
> More information about the other options on this screen can be found in Database Configuration Type in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard*

Task 6 Specifying JDBC Component Schema Information
Verify that the values on the JDBC Component Schema screen are correct for all schemas. If you selected **RCU Data** on the previous screen, then the schema table should already be populated appropriately.

> **Tip:**
>
> More information about the other options on this screen can be found in JDBC Component Schema in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*.

Task 7 Testing the JDBC Connections
Use the JDBC Test screen to test the datasource connections you have just configured.

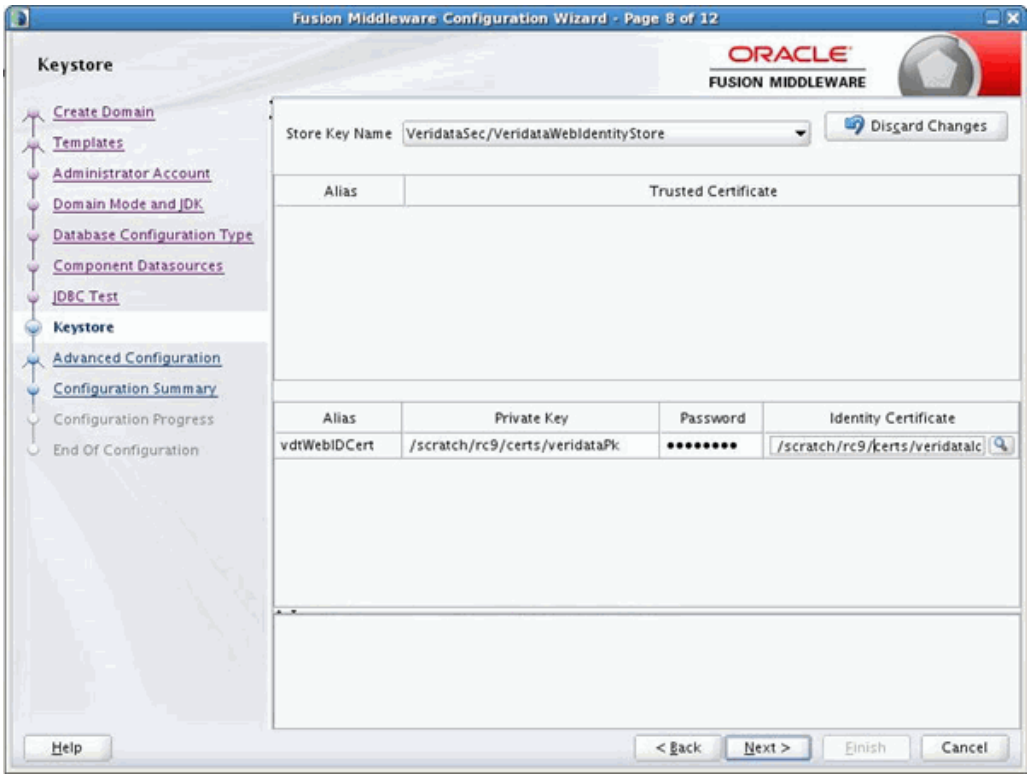
A green check mark in the Status column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

> **Tip:**
>
> More information about the other options on this screen can be found in JDBC Component Schema Test in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*

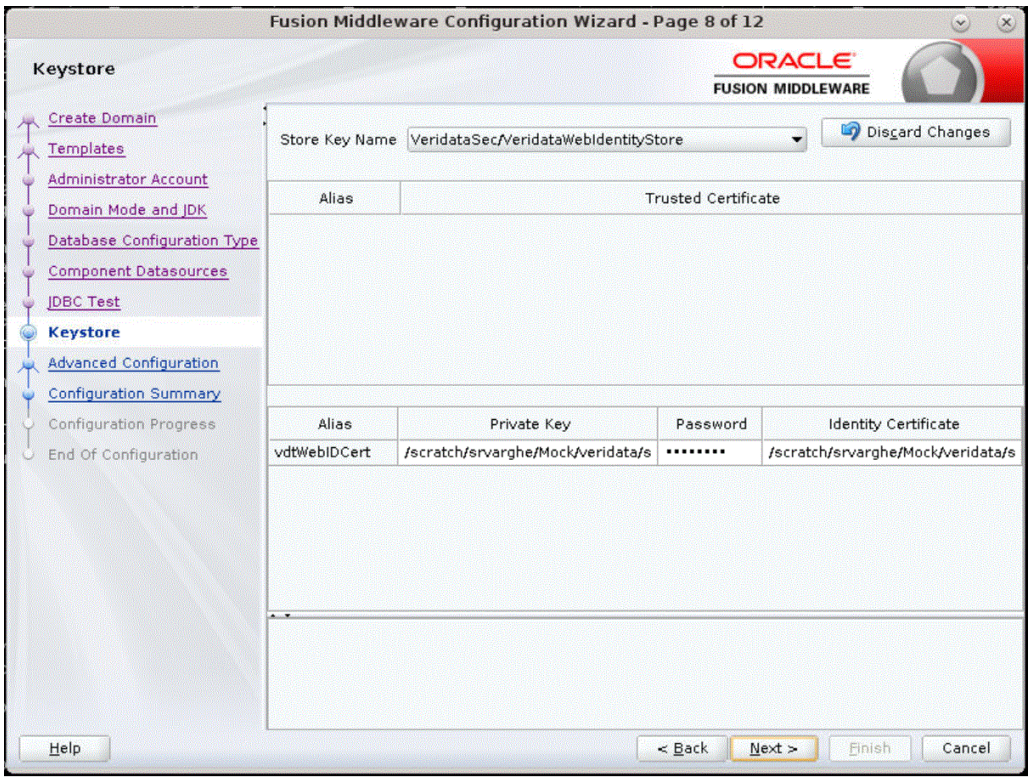
Task 8 Specifying Keystore Details

Use the Keystore screen to specify the path to the trusted certificate for each keystore. You can also specify the path to each keystore's private key, the password for the private key and the path to the Identity Certificate for the private key.

To select an Identity certificate, select the Identity keystore name from the Store Key Name drop down list and specify the paths to the private key and the identity certificate.



When you select the Trust Store from the Store Key Name drop-down list, the path to the sample certificate provided with the Veridata installation is displayed by default. You can specify the path to your trusted certificate on this page.



> **Tip:**
>
> More information about the other options on this screen can be found in JDBC Component Schema Test in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*

Task 9 Selecting Advanced Configuration
To complete domain configuration for the topology, select the following options on the Advanced Configuration screen:

- Administration Server

  This is required to properly configure the listen address of the Administration Server.
- Node Manager

  This is required to configure Node Manager.
- Managed Server, Clusters and Coherence

  This is required to configure the Oracle GoldenGate Veridata Managed Server.

Task 10 Configuring the Administration Server Listen Address
On the Administration Server screen, select the drop-down list next to Listen Address and select the IP address on the host where the Administration Server will reside. Do not use "All Local Addresses."
Do not specify any server groups for the Administration Server.

Task 11 Configuring Node Manager
The Node Manager screen can be used to select the type of Node Manager you want to configure, along with the Node Manager credentials.
Select **Per Domain** as the Node Manager type, then specify the Node Manager credentials.

> **Tip:**
>
> More information about the options on this screen can be found in Node Manager in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*.
> More information about the types of Node Manager can be found in Node Manager Overview in Administering Node Manager for Oracle WebLogic Server.

Task 12 Configuring Managed Servers
On the Managed Servers screen, a new Managed Server named `VERIDATA_server1` is created:

1. In the Listen Address drop-down list, select the IP address of the host on which the Managed Server will reside. Do not use "All Local Addresses."
2. In the Server Groups drop-down list, select **VERIDATA-MANAGED-SERVERS**. This server group ensures that the Oracle JRF services are targeted to the Managed Servers you are creating.

   Server groups target Fusion Middleware applications and services to one or more servers by mapping defined application service groups to each defined server group. A given application service group may be mapped to multiple server groups if needed. Any application services that are mapped to a given server group are automatically targeted to all servers that are assigned to that group. For more information, see Application Service Groups, Server Groups, and Application Service Mappings in *Domain Template Reference*.

These server names and will be referenced throughout this document; if you choose different names be sure to replace them as needed.

> **Note:**
>
> You must make a note of the IP address and port number for the Managed Server. You will need this information when you configure the agent.

> **Tip:**
>
> More information about the options on this screen can be found in Manged Servers in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*.

Task 13 Reviewing Your Configuration Specifications and Configuring the Domain
The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.
You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane.

Domain creation will not begin until you click **Create**.

> **Tip:**
>
> More information about the options on this screen can be found in Configuration Summary in the *Oracle Fusion Middleware Creating WebLogic Domains Using the Configuration Wizard Guide*.

Task 14 Reviewing Your Domain Home and Administration Server URL
The Configuration Success screen will show the Domain home location and URL of the Administration Server you just configured.

You must make a note of both items as you will need them to start the servers in your domain.

Click **Finish** to dismiss the configuration wizard.

5.2.3 Configuring the WebLogic Server Domain in a Silent Mode

When you install the Oracle GoldenGate Veridata, you need to configure the WebLogic Server domain for the Veridata server. In the Oracle GoldenGate Veridata, you can configure the domain in a silent or an offline mode.

To configure the WebLogic Server domain:

1. Copy the `domain_silent.properties` file from `$ORACLE_HOME/veridata/samples` to any location on the server and edit the properties file. The content of the properties file is as follows:

   ORACLE_HOME
   Path of the `Fusion Middleware/Oracle Home` directory.

   DOMAIN_LOCATION
   Path where new domain needs to be created. Specify along with domain name. For example, `/scratch/user_projects/domains/base_domain`.

   DB_HOST
   Enter the name of the server hosting the database.

   DB_PORT
   Enter the port number on which, the database listens.

   DB_SERVICE
   Enter the Database service name.

   SCHEMA_PREFIX
   Specify the prefix provided during schema creation in the step described in Navigating the RCU Screens to Create the Schema.

   WEBLOGIC_SERVER
   Hostname where the WebLogic Server is deployed.

   WEBLOGIC_PORT
   Port number on which, the WebLogic Server listens. For example, `7001`.

   VERIDATA_SERVER
   Hostname where Oracle GoldenGate Veridata is deployed.

VERIDATA_PORT
Port on which the Oracle GoldenGate Veridata will listen. For example, `8830`.

USE_SECURE_PASSWORD
This key specifies whether the password should be stored in file or read from command line. Value can be either of the following: `y` or `n`. When the value is set to `y`, the password is read from command line. When set to `n`, ensure to provide the password in this property file.

Ensure to update the following keys only when the `USE_SECURE_PASSWORD` key is set to `n`:

WEBLOGIC_USER
User name of the WebLogic application.

WEBLOGIC_PWD
Password of the Weblogic user to access the Weblogic application.

SCHEMA_PASSWORD
Password of the database schema given in the Creating the Oracle GoldenGate Veridata Repository Schema step.

USE_COMMON_SCHEMA_PASSWORD
This key specifies whether the password for all 5 schemas created in theCreating the Oracle GoldenGate Veridata Repository Schema step. should be same or not. While creating schemas if the password is set same for all 5 schemas then set the value of this key as y otherwise it can be set to `n`.

Ensure to update the following keys only when only when both `USE_SECURE_PASSWORD` and `USE_COMMON_SCHEMA_PASSWORD` are set to `n`:

VERIDATA_PWD
Specify the password for the `<SCHEMA_PREFIX>_VERIDATA` schema in Creating the Oracle GoldenGate Veridata Repository Schema step.

IAU_APPEND_PWD
Specify the password for the `<SCHEMA_PREFIX>_IAU_APPEND` schema in Creating the Oracle GoldenGate Veridata Repository Schema step.

OPSS_PWD
Specify the password for the `<SCHEMA_PREFIX>_OPSS` schema in Creating the Oracle GoldenGate Veridata Repository Schema step.

STB_PWD
Specify the password for the `<SCHEMA_PREFIX>_STB_PWD` schema in Creating the Oracle GoldenGate Veridata Repository Schema step.

IAU_VIEWER_PWD
Specify the password for the `<SCHEMA_PREFIX>_IAU_VIEWER_PWD` schema in Creating the Oracle GoldenGate Veridata Repository Schema step.

2. Save the properties file.
3. Execute the `domain_silent.sh` shell script. To do this, navigate to `$ORACLE_HOME/veridata/bin` and execute `domain_silent.sh` passing the `domain_silent.properties` file as an argument. For example, `./domain_silent.sh <property file>`.

If you want to automate the process of domain configuration, then you can do by editing the `domain_silent.properties` properties file only once with all necessary details and use the same file for script execution. In this case, there is no manual intervention needed after the property file has all the information available. However, you need to store the Weblogic and database credentials in the file.
If you don't want to store credentials in the properties file, then you can do that by setting the key `SECURE_PASSWORD=y` in the properties file. In this case, you need to enter the password using the command line when the script is triggered.

5.2.4 Updating the Oracle WebLogic Server Credentials

To update the Oracle WebLogic Server credentials:

1. Navigate to `$Oracle_Home/oracle_common/common/bin/` directory and run the following command: `./wlst.sh` to display the `wls:/offline>` prompt.
2. Run the following command to connect to the Admin server: `connect('<admin_user_name>','<admin_CURRENT_pwd>','<admin_console_url>')`. For example, run the `connect('admin','welcome2','localhost:7001/console')` command to display the following output:

   ```
   Connecting to t3://localhost:7001/console with userid admin ...
   Successfully connected to Admin Server "AdminServer" that belongs to
   domain "base_domain".
   ```

3. Run the following command: `cd('/SecurityConfiguration/<DOMAIN_NAME>/Realms/myrealm/AuthenticationProvide rs/DefaultAuthenticator')`. Update the *<DOMAIN_NAME>*.
4. Update the password using the following command: `cmo.resetUserPassword('<admin_user_name>','admin_NEW_pwd')`. For example, `cmo.resetUserPassword('admin','welcome3')`.

5.3 Performing Next Steps

Next Steps After Configuring an Oracle GoldenGate Veridata Domain contains basic administration tasks. You should familiarize yourself with the tasks described in this section and perform them as needed to verify that your domain is properly configured.

6

Deploying and Configuring Oracle GoldenGate Veridata Agent

Learn how to deploy and configure the Oracle GoldenGate Veridata agent of the Oracle WebLogic Server domain. Find out how to start and stop the agent, reload the logging properties, and how to use the agent in debugging mode.

6.1 About Oracle GoldenGate Veridata Agent Deployment Scripts

After installing Oracle GoldenGate Veridata Agent using Oracle Universal Installer, you must deploy the agent to a non Oracle Home location and configure the agent before running comparison jobs using a deployment script provided in the installation.

The following table describes the directories and the variables that are used when referring to those directories in this section.

Table 6-1 Directories in an Oracle GoldenGate Veridata Installation

Directory Variable	Directory Path
ORACLE_HOME	*/install_location*/Oracle_Home. This is the home directory for the Oracle Fusion Middleware products.
VERIDATA_PRODUCT_HOME	/*ORACLE_HOME*/veridata
AGENT_ORACLE_HOME	/*VERIDATA_PRODUCT_HOME*/agent
AGENT_DEPLOY_LOCATION	This is the location where the Oracle GoldenGate Veridata Agent is deployed. Note that this location should be outside the *ORACLE_HOME*.

The agent_config.sh|.bat script located in the *AGENT_ORACLE_HOME* directory is used for deploying the Veridata Agent.

Syntax:

```
agent_config.sh|.bat AGENT_DEPLOY_LOCATION
```

The *AGENT_DEPLOY_LOCATION* can be an absolute path or a path relative to the location from where the script is running.

> **Note:**
>
> You must deploy the Oracle GoldenGate Veridata Agent to a directory outside *ORACLE_HOME*.

6.2 Configuring Oracle GoldenGate Veridata Agent

You must configure the Oracle GoldenGate Veridata Agent to use your database.

1. Go to the agent deployed location *AGENT_DEPLOY_LOCATION*.

   There is an `agent.properties.sample` file in this directory that contains database related properties like JDBC URL and driver.

2. Copy the `agent.properties.sample` file and rename to `agent.properties`.

3. Most of the properties defined in the file have default values. However, you must update the following properties:

   a. The `server.port` property is the port where the Oracle GoldenGate Veridata Agent listens for connection requests.

      `server.port=`*`server.port`*

      For example, `server.port=7862`.

   b. The `database.url` specifies the JDBC connection URL for the database. Samples for all supported databases are provided in the file.

      `database.url=`*`database.url`*

      For example, `database.url=jdbc:oracle:thin:@localhost:1521:orcl`.

   c. The `server.jdbcDriver` property specifies the list of JDBC driver JAR files. Sample lists for the supported databases are provided in the file.

      Use the driver corresponding to the database.url in the preceding step.

      `server.jdbcDriver=<server.jdbcDriver>`

      For example, `server.jdbcDriver=ojdbc8.jar`.

4. Copy the drivers to the driver location which is available at: `AGENT_DEPLOY_LOCATION/drivers`.

6.3 Starting the Oracle GoldenGate Veridata Agent

Go to the agent deployment location *AGENT_DEPLOY_LOCATION* and run following command.

```
$ ./agent.sh start agent.properties
```

Where *agent.properties* is the properties file that contains your database properties.

6.4 Using the Oracle GoldenGate Veridata Agent Deployment Script in Debug Mode

For debugging issues with the Veridata Agent deployment, run the `agent_config` script with an additional command line argument as follows:

Syntax:

```
agent_config.sh|.bat AGENT_DEPLOY_LOCATION true
```

When this option is true, the debug logs are printed on the screen.

6.5 Reload the Logging Properties of the Veridata Agent

You can reload logging information from the `AGENT_ORACLE_HOME/config/odl.xml` configuration file to a running agent by using the `reloadLog` option. The changes in the `odl.xml` file are put into effect on the agent. The agent must be running for this command to work.

1. Open the command prompt and navigate to the directory where the agent is installed.
2. Run the following command to load the `AGENT_ORACLE_HOME/config/odl.xml` configuration file:

```
agent.sh|bat reloadLog
```

7

Running Oracle GoldenGate Veridata as a Windows Service

This topic describes how to run Oracle GoldenGate Veridata as a Windows Service.

7.1 Running Oracle GoldenGate Veridata as a Windows Service (for Server)

Locate the Command Scripts

1. Locate the `windows_service_admin.cmd_sample` and `windows_service_veridata.cmd_sample` files in the Oracle GoldeGate Veridata Installation directory. For example, `C:\Oracle\OracleHome122140\veridata\scripts`.
2. Edit both the files as per the instruction in the respective files and rename them by removing the `_sample` from the file names.
3. Copy the files to `ORACLE_HOME` directory.

> **Note:**
>
> Ensure that the Windows user has Read/Write permission on Oracle GoldenGate Veridata installed file.

Create Boot Identity File

1. If the boot identity file is not created for Weblogic to manage weblogic username and password, then follow the sections Creating a Boot Identity File for an Administration Server and Creating Boot Identity Files for Managed Servers in *Administering Server Startup and Shutdown for Oracle WebLogic Server Guide*.

> **Note:**
>
> For default installation of Oracle GoldenGate Veridata, Verdiata Managed Server uses the same boot identity file as the Admin Server. For Veridata Managed Server create `boot.properties` in the `Security` folder. For example, `C:\Oracle\OracleHome122140\user_projects\domains\base_domain\servers\VERIDATA_server1\security`

2. Locate the `cwallet.sso` file and check the appriopriate user permission (the user who will install the service should have Read/Write permission on this file). An example location path of the `cwallet.sso` can be: `C:\Oracle\OracleHome122140\user_projects\domains\base_domain\config\fmwconfig\bootstrap`

Install Admin Service

1. Locate `installSvc.cmd` in the Oracle WebLogic installed directory. For example, if `ORACLE_HOME` = `C:\Oracle\OracleHome122140`, then the command script will be located in `C:\Oracle\OracleHome122140\wlserver\server\bin`.
2. Take a backup of the script file.
3. Edit the script file before the tag `ENDLOCAL` and save the file.
   Remove the following lines:

```
"%WL_HOME%\server\bin\%WLSVC_EXE_TO_USE%" -install -svcname:"wlsvc
    %DOMAIN_NAME%_%SERVER_NAME%" -svcdescription:"%SERVICE_DESCRIPTION%"
    -javahome:"%JAVA_HOME%" -execdir:"%USERDOMAIN_HOME%"
    -maxconnectretries:"%MAX_CONNECT_RETRIES%" -host:"%HOST%" -port:"%PORT%"
    -extrapath:"%EXTRAPATH%" -password:"%WLS_PW%" -cmdline:%CMDLINE%
```

   Add the following lines:

```
"%WL_HOME%\server\bin\%WLSVC_EXE_TO_USE%" -install -svcname:"wlsvc
    %DOMAIN_NAME%_%SERVER_NAME%" -delay:180000  -
svcdescription:"%SERVICE_DESCRIPTION%"
    -javahome:"%JAVA_HOME%"  -execdir:"%USERDOMAIN_HOME%"
    -maxconnectretries:"%MAX_CONNECT_RETRIES%"  -host:"%HOST%" -port:"%PORT%"
    -extrapath:"%EXTRAPATH%"  -password:"%WLS_PW%" -cmdline:%CMDLINE%
    -log:"C:\Oracle\OracleHome122140\AdminServer-stdout.txt"
```

   In the above line, edit:
   `-svcname`: installed service name

   `-delay`: (in milliseconds) as per the startup time of the Oracle GoldenGate Veridata Server

   `-log`: Edit the location of the log file as per the install directory.
4. In the command prompt, run the `windows_service_admin.cmd` file.
5. Run `services.msc` to verify the installation.
6. Start the service and verify Admin Server is up (open the console UI in the browser).

For more information, see Setting Up a WebLogic Server Instance as a Windows Service.

Install Veridata Managed Service

1. Locate `installSvc.cmd` in the Oracle WebLogic installed directory. For example, if `ORACLE_HOME` = `C:\Oracle\OracleHome122140`, then the command script will be located in `C:\Oracle\OracleHome122140\wlserver\server\bin`.
2. Take a backup of the script file.
3. Edit the script file before the tag `ENDLOCAL` and save the file.
   Remove the following lines:

```
"%WL_HOME%\server\bin\%WLSVC_EXE_TO_USE%" -install -svcname:"wlsvc
    %DOMAIN_NAME%_%SERVER_NAME%" -svcdescription:"%SERVICE_DESCRIPTION%"
    -javahome:"%JAVA_HOME%" -execdir:"%USERDOMAIN_HOME%"
    -maxconnectretries:"%MAX_CONNECT_RETRIES%" -host:"%HOST%" -port:"%PORT%"
    -extrapath:"%EXTRAPATH%" -password:"%WLS_PW%" -cmdline:%CMDLINE%
```

   Add the following lines:

```
"%WL_HOME%\server\bin\%WLSVC_EXE_TO_USE%" -install -svcname:"wlsvc
    %DOMAIN_NAME%_%SERVER_NAME%" -delay:180000 -depend:"wlsvc
base_domain_AdminServer"
```

```
    -svcdescription:"%SERVICE_DESCRIPTION%"  -javahome:"%JAVA_HOME%"
    -execdir:"%USERDOMAIN_HOME%"  -maxconnectretries:"%MAX_CONNECT_RETRIES%" -
host:"%HOST%"
    -port:"%PORT%"  -extrapath:"%EXTRAPATH%" -password:"%WLS_PW%" -cmdline:%CMDLINE%
    -log:"C:\Oracle\OracleHome122140\VeridataServer-stdout.txt"
```

In the above line, edit:
`-svcname`: installed service name

`-delay`: (in milliseconds) as per the startup time of the Oracle GoldenGate Veridata Server

`-depend`: Admin Server service name

`-log`: Edit the location of the log file as per the install directory.

4. In the command prompt, run the `windows_service_veridata.cmd` file.
5. Run `services.msc` to verify the installation.
6. Start the service and verify that the Oracle GoldenGate Veridata Server is up (open the console UI in the browser).

> **Note:**
>
> Before starting the Oracle GoldeGate Veridata Server service, ensure that the Admin Server is up and running.

For more information, see Requiring Managed Servers to Start After Administration Servers.

7.2 Running Oracle GoldenGate Veridata as a Windows Service (for Agent)

On Windows platform, after installing the Oracle GoldenGate Veridata Agent, run the agent as a service. The scripts are located in *AGENT_DEPLOY_LOCATION*\service. The `README.txt` file in the `service` directory describes the procedure to run the agent as a service.

To install as a Windows service:

1. Go to the Oracle GoldenGate Veridata Agent deployment directory.
2. Navigate to the **Service** folder.
3. Edit the `agent_service.properties` file.
4. Find `_VERIDATA_SERVICE_NAME` and change it to the service name for the agent. The name can only contain alphabets or numbers. No special character or white space is allowed.
5. Find `_AGENT_PROPERTY_FILE` and change it to the agent properties file name which has to be run as a service.
6. Save and close the file.
7. Run a Windows command session.
8. Run the `Install_veridata_Agent.bat` script.

To uninstall the agent service:

1. Run a Windows command session.
2. Execute `Uninstall_Veridata_Agent.bat` from the deployment directory of the agent.

To start and stop the Veridata agent:

1. Open the command prompt and navigate to the directory where the agent is deployed.
2. Run one of the following commands:

   (starts the agent)

   ```
   Start_Veridata_Agent.bat
   ```

   or

   ```
   Stop_Veridata_Agent.bat
   ```

8

Configuring Oracle GoldenGate Veridata Agent Using Kerberos to Connect to Oracle Database

To configure Oracle GoldenGate Veridata Agent using Kerberos to connect to Oracle database:

1. Complete the steps detailed in Deploying and Configuring Oracle GoldenGate Veridata Agent.
2. Initiate initial ticket granting ticket for the principal using `okinit`. To request an initial ticket, run `okinit username`. The "username" is the user created or configured to use kerberos.
3. Login database instance with an **Oracle Net Service** service name. Run `sqlplus / @service_name` to login to the db instance, and then run `show user`. The displayed user should be the user granted the initial ticket before.
4. Copy Kerberos configuration file and ticket cache file into Veridata agent deploy directory. Absence of either file in agent deploy directory disables the kerberos use of the Oracle GoldenGate Veridata agent.
5. Edit `agent.properties.oracle`. For example: `database.url=jdbc:oracle:thin:@host1.us.oracle.com:1522:vdtkbr`. The database.url is the same as the url that is in a non-kerberos configuration.
6. In the `agent.properties.oracle` file, add, uncomment, and edit the entries, `kerberos.configuration.file.name` and `oracle.kerberos.ticket.cache.file.name`. A missing entry or an incorrect entry disables the kerberos use of Veridata agent. If kerberos use is not desired, then comment out or delete either of the entries.

   For example:

   ```
   #Kerberos configuration file name.
   Comment the entry to disable veridata agent to use kerberos.
   #To make veridata agent to use kerberos, the file must be in the agent install
   directory.
   kerberos.configuration.file.name=krb.conf
   #Kerberos ticket cache file name for Oracle.
   #To make veridata agent to use kerberos, the file must be in the agent install
   directory.
   oracle.kerberos.ticket.cache.file.name=krb.cc
   ```
7. Start the Veridata Agent: `./agent.sh`.
8. Verify connection in UI. Note that you do not have to enter the username and password in the Database details.

9

Configuring Oracle GoldenGate Veridata Agent Using Kerberos to Connect to Hive

To configure Oracle GoldenGate Veridata Agent using Kerberos to connect to Hive database:

1. Complete the steps detailed in Deploying and Configuring Oracle GoldenGate Veridata Agent.
2. Obtain the Kerberos configuration file from the kerberos server, for example, `krb5.conf`.
3. Copy the Kerberos configuration file in the OS default location. For example, in Linux, it is `/etc/`.
4. For Hive specific configurations, review `MIDDLEWARE_HOME/veridata/agent/sample_properties/agent.properties.hive`.
5. Obtain the the keytab file from Kerberos enabled Hive environment.
6. Copy the keytab file into the Oracle GoldenGate Veridata Agent deploy directory.
7. In the `agent.properties` file, add/uncomment, and edit the entries `hive.kerberos.keytab.file.name` and enter the keytab file name.
8. Edit `agent.properties` and add the `database.url`. For Kerberos authentication principal argument is required . For example, in a Cloudera Hive following is the jdbc url: `syntaxdatabase.url=jdbc:hive2://HiveServerHost:10000/default;principal=hive/HiveServerHost@YOUR-REALM.COM`.
9. Edit `agent.properties.hive` and add the `server.jdbcDriver` appropriately. For example, for Cloudera Hive:

   ```
   server.jdbcDriver=commons-collections-3.2.1.jar hadoop-common-2.4.1.jar
   hive-service-0.14.0.jar hadoop-mapreduce-client-core-2.4.1.jar hive-shims-
   common-0.14.0.jar
   commons-logging-1.1.3.jar hive-exec-0.14.0.jar log4j-1.2.17.jar hive-jdbc-0.14.0-
   standalone.jar
   slf4j-api-1.7.5.jar hadoop-auth-2.4.1.jar hive-metastore-0.14.0.jar slf4j-
   log4j12-1.7.5.jar
   commons-configuration-1.6.jar commons-dbcp2-2.5.0.jar commons-pool2-2.6.0.jar
   ```

   Obtain the appropriate versions of these jars from Hive environment.
10. Initiate an initial ticket granting ticket for the principal using `kinit`. Go to the agent installation directory and run `kinit` and verify using:

    ```
    klist: kinit
    -k -t  {keytab file}  {principal name}
    ```

> **Note:**
>
> For auto renewal of Kerberos ticket, add the `hive.kerberos.principal.override` property in the `agent.properties` file as follows: `hive.kerberos.principal.override=<Principal name>`. If you do not mention the Principal name, then Oracle GoldenGate Veridata uses the principal name form the `database.url`.

11. Start the Oracle GoldenGate Veridata Agent.
12. Verify connection in UI.

> **Note:**
>
> In case you have permission-related issues, you may have to enter the Hadoop username and password in Database details. First, try without using the username and password.

10

Connecting Oracle GoldenGate Veridata to SSL-Enabled Oracle Database

SSL Enabled JDBC URL format: `jdbc:oracle:thin:@tcps://<host>:<port>/servicename?wallet_location=<wallet path>`, where `wallet_location` is the Directory path for client wallet files, which are `cwallet.sso` and `ewallet.p12`.

To connect Oracle GoldenGate Veridata to SSL-Enabled Oracle Database:

1. Create an auto-login wallet in the database as follows: `$ orapki wallet create -wallet <wallet path> -pwd <wallet password> -auto_login`.
2. Create a self-signed certificate and load it into the wallet: `$ orapki wallet add -wallet <wallet path> -pwd <wallet password> -dn "CN=<database hostname>" -keysize 1024 -self_signed -validity 3650`
3. Export the certificate so that you can load it into the client wallet: `$ orapki wallet export -wallet <wallet path> -pwd <wallet password> -dn "CN=<hostname>" -cert <server certificate path>`
4. Repeat step 1 to 3.
5. Exchange Client and Server Certificates:
    a. Load the server certificate into the client wallet: `$ orapki wallet add -wallet <client wallet path> -pwd <wallet password> -trusted_cert -cert <server certificate path>`
    b. Load the client certificate into the server wallet: `$ orapki wallet add -wallet <server wallet path> -pwd wallet password -trusted_cert -cert <client certificate path>`
6. Check the contents of the server/client wallet: `$ orapki wallet display -wallet <server wallet path> -pwd <wallet password>`

For more information, see Create JKS Wallets for a TLS Connection to a DB System that has Client Authentication Enabled in the *Administering Oracle Data Safe guide*.

11

Connecting Oracle GoldenGate Veridata to SSL-Enabled MySQL Database

Setting up Server Authentication via server certificate

1. Copy `ca.pem` file from MySQL database server to veridata agent machine.
2. Run the `keytool` command in the veridata agent machine to import the `ca.pem` file: `$> keytool -importcert -alias <Set_Your_Alias> -file ca.pem -keystore truststore -storepass <Set_Your_Password>`

> **Note:**
>
> If the truststore file does not already exist, then a new one is created; else the certificate gets added to the existing file.

3. Append the following connection parameters to `MySQL JDBC URL` in the `agent.properties` file. For MySQL version 8.0.12 and earlier: `database.url=jdbc:mysql://abc.com:3306?useSSL=true&verifyServerCertificate=true`. For MySQL version 8.0.13 and later: `database.url=jdbc:mysql://abc.com:3306?sslMode=<VERIFY_CA or VERIFY_IDENTITY>`.
4. Export java parameters to read the truststore you just created or modified. Export `JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStore=<path_to_truststore_file> -Djavax.net.ssl.trustStorePassword=<your_truststore_password>"`

Setting up Client Authentication via client certificate

1. Copy `client-cert.pem` and `client-key.pem` two files from MySQL database server to veridata agent machine.
2. Run the `openssl` command in the veridata agent machine to convert the client key and certificate files to a PKCS #12 archive: `$> openssl pkcs12 -export -in client-cert.pem -inkey client-key.pem -name "<Set_Your_name>" -passout pass:<Set_Your_Password> -out client-keystore.p12`
3. Run the `keytool` command in the veridata agent machine to import the PKCS file: `$> keytool -importkeystore -srckeystore client-keystore.p12 -srcstoretype pkcs12 -srcstorepass <Set_Your_Password> -destkeystore keystore -deststoretype JKS -deststorepass <Set_Your_Password>.`

> **Note:**
>
> If the keystore file does not already exist, then new one is created; else, the certificate is added to the existing file.

After the step, you can delete the PKCS #12 archive (client-keystore.p12 in the example).

4. Export java parameters to read the keystore you just created or modified: `export JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.keyStore=<path_to_keystore_file> -Djavax.net.ssl.keyStorePassword=<your_keystore_password>" .`
5. Authentication via client certificate does not require connection parameters in `MySQL JDBC URL` as opposed to via server certificate.

2-Way Authentication

Apply the steps outlined in both *Setting up Server Authentication via server certificate* and *Setting up Client Authentication via client certificate* topics.

12

Connecting Oracle GoldenGate Veridata to SSL-Enabled SQL Server Database

SSL SYNTAX: `database.url=jdbc:sqlserver://<host>:<port>;databaseName=<dbName>;encrypt=false;trustServerCertificate=false;`

Example URL: `:database.url=jdbc:sqlserver://phoenix007.dev3sub2phx.databasede3phx.oraclevcn.com:1433;databaseName=atssrc;encrypt=false;trustServerCertificate=false;`

To connect Oracle GoldenGate Veridata to SSL-EnabledSQL Server:

1. Import the mssql crt file to the truststore: `keytool -importcert -alias ms_sql -keystore $JAVA_HOME/jre/lib/security/cacerts -storepass changeit -file <path>/mssql.pem`.
2. Update the jdbc url in `agent.properties`.
3. Set the path of truststore and truststore password in `$JAVA_OPTS` variable `export JAVA_OPTS="$JAVA_OPTS -Djavax.net.ssl.trustStore=<path_to_truststore_file> -Djavax.net.ssl.trustStorePassword=<your_truststore_password>"`
4. Start the agent.

13

Connecting Oracle GoldenGate Veridata to SSL-Enabled PostgreSQL Database

SSL SYNTAX: `database.url=jdbc:postgresql://<machine_name>:5432/<db_name>?` **`sslmode=verify-ca&sslrootcert=<full path of the cert>/root.crt`**

Example URL: `database.url=jdbc:postgresql://phoenix007:5432/postgres?` **`sslmode=verify-ca&sslrootcert=<path>/root.crt`**

To connect Oracle GoldenGate Veridata to SSL-Enabled PostgreSQL:

1. Copy the root certificate root.crt in your agent location and provide full path of it for `sslrootcert` parameters.
2. For `sslmode` there can be two values `verify-ca` and `verify-full`.

14

Next Steps After Configuring an Oracle GoldenGate Veridata Domain

After you have successfully installed and configured Oracle GoldenGate Veridata, ascertain how to start your administration and manage servers, create users, and launch the Oracle GoldenGate Veridata Web User Interface.

This chapter includes the following topics:

14.1 Verifying the Oracle GoldenGate Veridata Installation

After you complete the installation, you can verify it by successfully completing the following tasks:

- Reviewing Installation Log Files
- Verifying the Directory Structure

14.1.1 Reviewing Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see Configuring Installation and Configuration Log Files in I*nstalling with the Oracle Universal Installer*.

14.1.2 Verifying the Directory Structure

Review the directory structure after installing and configuring Oracle GoldenGate Veridata. For more information about the directory structure you should see after installation, see Understanding the Oracle GoldenGate Veridata Directory Structure.

14.2 Starting the Servers

After configuration is complete, do the following to access the tools with which you can manage your domain:

> **Note:**
>
> For more information on additional tools you can use to manage your domain, see Overview of Oracle Fusion Middleware Administration Tools in *Oracle Fusion Middleware Administering Oracle Fusion Middleware*.

- Starting the Administration Server
- Starting the Managed Server using Veridata Scripts

14.2.1 Starting the Administration Server

To start the Administration Server, go the *DOMAIN_HOME*/bin directory.

On UNIX operating systems, run:

```
./startWebLogic.sh
```

On Windows operating systems, run:

```
startWebLogic.cmd
```

If you selected **Production Mode** on the Domain Mode and JDK screen in Task 4, Specifying the Domain Mode and JDK, you will be prompted for the login credentials of the Administrator user as provided on the Administrator Account screen in Task 3, Configuring the Administrator Account.

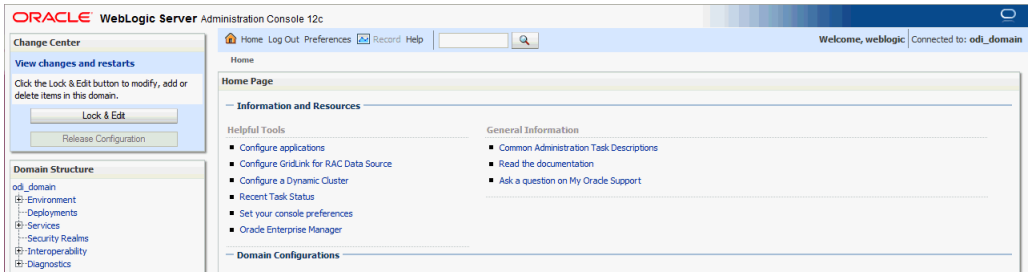
> **Tip:**
>
> For more information about starting the Administration Server, see Starting and Stopping Administration Server in *Oracle Fusion Middleware Administering Oracle Fusion Middleware Guide*.
>
> In production mode, a boot identity file can be created to bypass the need to provide a user name and password when starting the Administration Server. For more information, see Creating a Boot Identity File for an Administration Server in *Administering Server Startup and Shutdown for Oracle WebLogic Server*.

You can verify that the Administration Server is up and running by access the Administration Server Console. The URL is provided on the Configuration Success screen in Task 14, Reviewing Your Domain Home and Administration Server URL.

> **Note:**
>
> Make sure that the database hosting your product schemas is up and running and accessible by the Administration Server.

```
http://administration_server_host:administration_server_port/console
```

The default Administration Server port number is `7001`.



For more information about how to use the Administration Console, see Getting Started Using Oracle WebLogic Server Administration Console in *Administering Oracle Fusion Middleware*.

14.2.2 Starting the Managed Server using Veridata Scripts

The Veridata installation provides you with a script, `veridataServer(.sh/.bat)`, to start and stop the Veridata Managed Server. These scripts are located in the *DOMAIN_HOME*`/veridata/bin` folder.

> **Note:**
>
> here is no Managed Server in a compact domain. Starting the Administration Server will start the entire domain.

To run the `veridataServer` script:

1. On the system where the Oracle GoldenGate Veridata is installed, run the command shell of the operating system.
2. Navigate to the *DOMAIN_HOME*`/veridata/bin` directory.
3. Use the following syntax to run the `veridataServer` script.

Example 14-1 Syntax

```
veridataServer{.bat|.sh} start/stop [SERVER_NAME] [ADMIN_URL]
```

Optional Parameters	Description
SERVER_NAME	Specifies the name of the Veridata Managed Server that you want to start or stop. If not specified, the script starts the default managed server (`VERIDATA_server1`).
ADMIN_URL	Specifies the URL for the Veridata domain administration server. Default URL is `t3://<admin server>:<port>`.

14.3 Creating Oracle GoldenGate Oracle GoldenGate Veridata Users and Assigning Privileges

After you configure the Oracle GoldenGate Veridata domain, you must create users and assign privileges for these users to access the Oracle GoldenGate Veridata Web User Interface. Use Oracle WebLogic Administration Console to create users.

By default, the WebLogic Domain administrator has privileges to access the Veridata application.

For more information about creating users and groups by using Administration Console, see "Create Users" in *Oracle WebLogic Server Administration Console Online Help*.

To know more about the user roles for Veridata, see Getting Started Using Oracle WebLogic Server Administration Console in *Administering Oracle GoldenGate Veridata*.

14.4 Launching Oracle GoldenGate Veridata Web User Interface

To launch Oracle GoldenGate Veridata web application, enter the following URL in a web browser:

```
http://hostname:8830/veridata/
```

where *hostname* represents the host where you have installed the Oracle GoldenGate Veridata Server.

To log in to the Oracle GoldenGate Veridata application, use the Oracle GoldenGate Veridata User credentials specified in the Administration Console. See "Creating Oracle GoldenGate Oracle GoldenGate Veridata Users and Assigning Privileges".

15
Deinstalling Oracle GoldenGate Veridata

We recommend that you always use these instructions to remove the software. Note that if you try to remove the software manually, you may encounter problems when you try to reinstall the software again.
This chapter includes the following sections:

15.1 Understanding Product Deinstallation

The Oracle Fusion Middleware deinstaller removes the software from the Oracle home directory from which it is started. Table 15-1 summarizes the procedure and provides links to supporting documentation.

Table 15-1 Roadmap for Deinstalling GoldenGate Veridata

Task	Description	Documentation
Stop Oracle Fusion Middleware.	All servers and processes in your domain should be stopped before running the deinstaller.	See Stopping Oracle Fusion Middleware.
Remove your database schemas.	Run Repository Creation Utility to remove your database schemas.	See Removing Your Database Schemas.
Remove the software.	Run the product deinstaller to remove Oracle Fusion Middleware Infrastructure.	See Deinstalling the Software.
Remove the Oracle home directory.	The deinstaller does not remove all files and folders from the Oracle home directory. After the deinstaller is finished, you must manually remove the Oracle home to complete your product removal.	See Removing the Oracle Home Directory Manually.
Remove your domain and application data.	The deinstaller does not remove data contained in your Domain home or Application home directories, even if they are located inside the Oracle home. You must remove these directories manually.	See Removing Your Domain and Application Data.
Remove Oracle Fusion Middleware Infrastructure.	In you created the topology for Java EE agents, you can also remove the Oracle Fusion Middleware Infrastructure software from your system.	See Deinstalling Oracle Fusion Middleware Infrastructure.

15.2 Stopping Oracle Fusion Middleware

Stopping Oracle Fusion Middleware involves the following:

- Stopping Your Oracle GoldenGate Veridata Agent
- Stopping the Oracle Fusion Middleware Environment

15.2.1 Stopping Your Oracle GoldenGate Veridata Agent

Before running the deinstaller, you should stop all agents that are running.

To stop your standalone agent, go to the Veridata Agent deployment directory and run the following commands:

On UNIX operating systems:

```
./agent.sh stop
```

On Windows operating systems:

```
./agent.bat stop
```

15.2.2 Stopping the Oracle Fusion Middleware Environment

Before running the deinstaller, you should stop all servers and processes associated with the Oracle home you are going to remove.

For more information, see Stopping an Oracle Fusion Middleware Environment in *Administering Oracle Fusion Middleware*.

15.3 Removing Your Database Schemas

Before your Oracle home is removed, you should run Repository Creation Utility to remove the database schemas associated with this domain. Each domain has its own set of schemas, uniquely identified by a custom prefix (see About Custom Prefixes in the *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*. This set of schemas cannot be shared with any other domain (see Planning Your Schema Creation in the *Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*).

If there are multiple sets of schemas on your database, be sure to identify the schema prefix associated with the domain you are removing.

For schema removal instructions, see Dropping Schemas in the*Oracle Fusion Middleware Creating Schemas with the Repository Creation Utility Guide*.

15.4 Deinstalling the Software

Follow the instructions in this section to start the product deinstaller and remove the software.

- Starting the Deinstaller
- Navigating the Deinstallation Screens

If you want to perform a silent (command-line) deinstallation, see Running the Oracle Universal Installer for Silent Deinstallation in *Installing Software with the Oracle Universal Installer*.

15.4.1 Starting the Deinstaller

To start the deinstaller on a Windows operating system, do one of the following:

1. Use a file manager window to go to the *ORACLE_HOME*\oui\bin directory and double click on deinstall.cmd.
2. From the command line, go to the *ORACLE_HOME*\oui\bin and enter the following command:

   ```
   deinstall.cmd
   ```

3. From the **Start** menu, select **All Programs**, then select **Oracle**, then select **OracleHome**, then select **Uninstall Oracle Middleware**.

To start the deinstaller on a UNIX operating system, go to the *ORACLE_HOME*/oui/bin directory and enter the following command:

```
./deinstall.sh
```

15.4.2 Navigating the Deinstallation Screens

The deinstallation program displays a series of screens, in the order listed in Table 15-2.

If you need additional help with any of the deinstallation screens, click the screen name.

Table 15-2 Deinstall Screens and Descriptions

Screen	Description
Welcome	This screen introduces you to the product deinstaller.
Deinstallation Summary	This screen shows the Oracle home directory and its contents that will be deinstalled. Verify that this is the correct directory. Click **Deinstall** to begin removing the software.
Deinstallation Progress	This screen shows the progress of the deinstallation.
Deinstallation Complete	This screen appears when the deinstallation is complete. Review the information on this screen, then click **Finish** to dismiss the deinstaller.

15.5 Removing the Oracle Home Directory Manually

After the deinstaller is finished, you must manually remove your Oracle home directory and any existing sub-directories that were not removed by the deinstaller. For example, if your Oracle home directory was `/home/oracle/product/Oracle_Home` on a UNIX operating system:

```
> cd /home/oracle/product
> rm -rf Oracle_Home
```

On a Windows operating system, if your Oracle home directory was `C:\Oracle\Product\Oracle_Home`, use a file manager window and navigate to the `C:\Oracle\Product` directory, then right-click on the `Oracle_Home` folder and select **Delete**.

> **Note:**
>
> You can delete the `Oracle_Home` directory only if Oracle GoldenGate Veridata was installed in it. If any other product is configured within the same location, then delete just the Oracle GoldenGate Veridata specific files from the directory.

15.6 Removing Your Domain and Application Data

To remove your domain and application data:

1. Manually remove your Domain home directory.

   For example, if your Domain home directory was `/home/oracle/config/domains/veridata_domain` on a UNIX operating system:

   ```
   > cd /home/oracle/config/domains
   > rm -rf veridata_domain
   ```

   On a Windows operating system, if your Domain home directory was `C:\Oracle\Config\domains\odi_domain`, use a file manager window and navigate to the

`C:\Oracle\Config\domains` directory, then right-click on the `odi_domain` folder and select **Delete**.

2. Manually remove your Application home directory.

   For example, if your Application home directory was `/home/Oracle/config/applications/veridata_domain` on a UNIX operating system:

   ```
   > cd /home/Oracle/config/applications
   > rm -rf veridata_domain
   ```

   On a Windows operating system, if your Application home directory was `C:\Oracle\Config\applications\veridata_domain`, use a file manager window and navigate to the `C:\Oracle\Config\applications` directory, then right-click on the `odi_domain` folder and select **Delete**.

3. Backup the `domain_registry.xml` file in your Oracle home, then edit the file and remove the line associated with the domain you are removing. For example, to remove the `odi_domain`, find the following line and remove it:

   ```
   <domain location="/home/Oracle/config/domains/veridata_domain"/>
   ```

   Save and exit the file when you are finished.

15.7 Deinstalling Oracle Fusion Middleware Infrastructure

After you have removed your Oracle GoldenGate Veridata software, you can choose to remove the Oracle Fusion Middleware Infrastructure from your machine. Before doing so, make sure there are no other products using the Infrastructure; those products will no longer function once the Infrastructure is removed.

For deinstallation instructions, see Deinstalling Oracle Fusion Middleware Infrastructure in *Installing and Configuring the Oracle Fusion Middleware Infrastructure*.

15.8 Reinstalling the Software

You can reinstall your software into the same Oracle home as a previous installation only if you have deinstalled the software according to the instructions in this chapter, including manually removing the Oracle home directory. When you reinstall, you can then specify the same Oracle home as your previous installation.

Consider the following cases where the Oracle home is not empty:

- Installing in an existing Oracle home that contains the same feature sets.

  The installer will warn you that the Oracle home you specified during the installation already contains the same software you are trying to install. Your options are to:

  1. Select a different installation type. In this case, only the feature sets that do not already exist in the Oracle home directory will be installed.
  2. Select a different Oracle home directory.

- Installing in an existing, non-empty Oracle home.

  For example, suppose that you have chosen to create your Domain home or Application home somewhere inside your existing Oracle home. This data is not removed during the deinstallation process, so if you attempt to reinstall into the same Oracle home, the installer will not allow it. Your options are to:

1. Deinstall your software from the Oracle home as described in this chapter, and then remove the Oracle home directory. After this is complete, you can reinstall and reuse the same Oracle home location, using the instructions in Installing Oracle GoldenGate Veridata. Any domain or application data that was in the Oracle home will have to be re-created.
2. Select a different Oracle home directory.

A

Installing Oracle GoldenGate Veridata C-Agent

Learn how to install the Oracle GoldenGate Veridata C-Agent on a UNIX/Linux, Windows, or NonStop platform.
This chapter includes the following sections:

A.1 Installation Overview

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

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.

A.2 Installing the C-Agent on a NonStop System

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

1. Install the Oracle GoldenGate Veridata Agent files.
2. Copy `VSNSERV` to remote nodes if they contain table partitions.
3. 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)
4. Configure the Manager process.
5. (Optional) Add the following parameter that will rollover report files after the specified increment

   ```
   VERIDATAREPORTAGE <nnn> [time units]
   ```

   Time units are, one of:

   DAY(s), HOUR(S), MINUTE(s), or SECOND(s)

   For example:

   ```
   VERIDATAREPORTAGE 1 DAY
   ```

A.2.1 Installing the Oracle GoldenGate Veridata Agent Files

1. Follow the steps in Oracle GoldenGate Veridata Distribution to download the Oracle GoldenGate Veridata Agent build file to a Windows workstation.
2. Using 7-zip, 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. Execute the following TACL command:

```
unpak VERJ06PK (or VERH06PK) , $*.*.*, <userid>, vol <volume>.<subvolume>, KEEP
```

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

```
Run veriinst
```

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.

A.2.2 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
FUP license <volume>.<subvolume>.VSNSERV
```

   - 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.

A.2.3 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 Veridata agent.
- Partitions for tables that will be compared with 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.

A.2.4 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
   ```

B

Understanding the Oracle GoldenGate Veridata Directory Structure

Understand the directory structure, which is created after the Oracle GoldenGate Veridata installation.
Note that the sample directories in this section illustrate the recommended directory structure, as described in "Understanding the Recommended Directory Structure" in *Planning an Installation of Oracle Fusion Middleware*. Your directory structure may differ slightly based on where you choose to create these directories on your system.

B.1 Oracle Home Directory Structure

Figure B-1 shows an example of the basic directory structure of your `ORACLE_HOME` after installation and domain configuration of Oracle GoldenGate Veridata in a WebLogic Server domain. Note that figure shows only important directories. There are other directories and files created too.

Figure B-1 Directory Structure of Oracle_Home

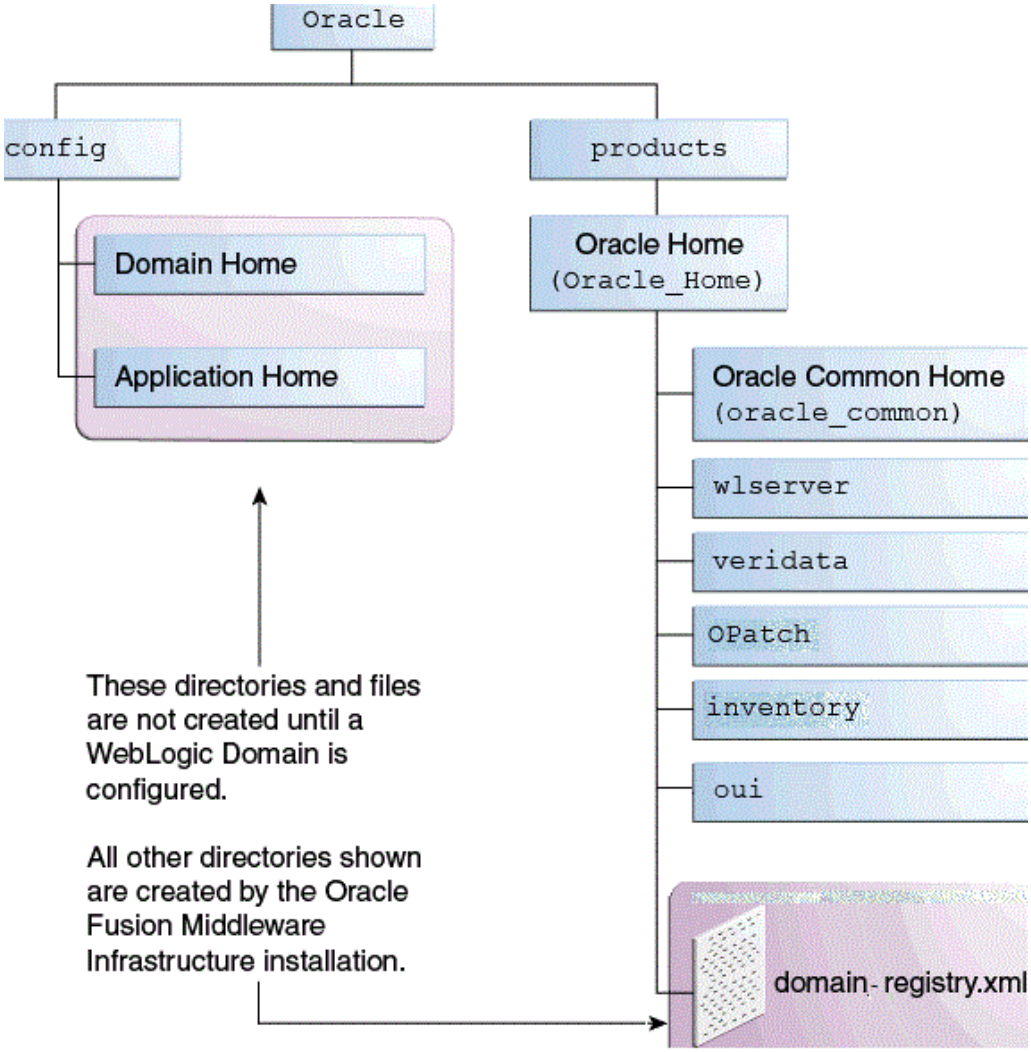


Table B-1 describes these directories in more details.

Table B-1 Directory Description for Oracle GoldenGate Veridata in a WebLogic Server Domain

Directory or file	Description
`oracle_common`	This directory contains the common binary and library files required for Oracle WebLogic Server.
`wlserver`	This directory contains a small portion of webLogic Server binaries that are used for management functionality. This directory is also referred to as the WebLogic home directory, or *`WL_HOME`*.

Table B-1 (Cont.) Directory Description for Oracle GoldenGate Veridata in a WebLogic Server Domain

Directory or file	Description
veridata	This directory contains the Oracle GoldenGate Veridata binaries and also the following directories: • `agent`: represents the home directory for the Veridata agent • `bin`: contains the scripts for the Veridata agent • `clilib`: contains the command-line library files for Veridata • `common`: contains Veridata domain templates • `plugin`: contains Veridata plug-in for Upgrade Assistant • `samples`: contains the sample certificates for SSL configuration • `t2p`: contains the scripts and instructions for moving from a test to production environment • `webapps`: contains the Veridata web application EAR file
`install`	This directory contains install-related files and scripts.
`inventory`	This directory contains information about the components, feature sets, and patches installed in this Oracle home directory.
`logs`	This directory contains the log files for your installation and configuration transactions.
`odi`	This directory contains the ODI binaries.
`OPatch`	This directory contains OPatch and supported files. OPatch is a tool used to patch Oracle Fusion Middleware software.
`oui`	This directory contains files used by the Oracle Universal Installer, including the deinstallation program. If you need to run the Oracle Universal Installer again after the product is installed, you should do so from this directory.
`domain-registry.xml`	This registry file contains the location of all domains currently registered with this WebLogic Server installation. Whenever you add a new domain, it is registered in this file.

B.2 Veridata Domain Home Directory Structure

Figure B-2 shows an example of the basic directory structure of your *`VERIDATA_DOMAIN_HOME`* after installation and domain configuration of Oracle GoldenGate Veridata in a WebLogic Server domain.

For more information about the domain home and the contents, see Domain Directory Contents in *Understanding Domain Configuration for Oracle WebLogic Server*.

Figure B-2 Veridata Domain Home directory structure

Table B-2 describes the important directories in the Oracle GoldenGate Veridata domain.

Table B-2 Directory Description for Oracle GoldenGate Veridata Domain

Directory or file	Description
`bin`	This directory contains scripts that are used in the process of starting and stopping the Administration Server and the Managed Servers in the domain.
config	This directory contains the current configuration and deployment state of the domain. The `veridata.cfg` configuration file is located in a sub directory at `config/veridata`.
`logs`	This directory contains the domain-level log files.
`servers`	This directory contains one subdirectory for each Oracle WebLogic Server instance in the domain.
`veridata`	This directory contains the following subdirectories: • bin: contains the Veridata command-line utilities. • reports: default location for the out-of-sync files and reports.

C

Oracle GoldenGate Veridata Distribution

Here we understand how to download the files of Oracle GoldenGate Veridata files; as well as C-Agent; and Java Agent for DB2 for i or DB2 z/OS.
This appendix includes the following sections:

C.1 Downloading Oracle GoldenGate Veridata

A distribution is an archive file containing an installer; when you run the installer, the set of Oracle GoldenGate Veridata components and feature that are included with the distribution are installed. You will need a certified JDK on your system in order to be able to run the installer from the `.jar` file. For more information, see Prerequisites for Installing Oracle GoldenGate Veridata .

Product distributions can be obtained from either the Oracle Software Delivery Cloud or from Oracle Technology Network. See Downloading the Product .

Table C-1 describes the products and feature sets in Oracle GoldenGate Veridata.

Table C-1 Oracle GoldenGate Veridata Product and Feature Sets

Product	Feature Set	Description
Oracle GoldenGate Veridata	Veridata Server	The Veridata Server component includes the web server, the web application, and command-line utilities such as `vericom` .
	Veridata Agent	This component installs only the Veridata Agent.
Internal Features	Apache Ant	This is a software tool used for automating build processes.
	FMW Upgrade	This is the upgrade assistant that can be used to upgrade your 11*g* Oracle GoldenGate Veridata software to 12*c*. For more information, see Upgrading Oracle GoldenGate Veridata.
	OPatch	The OPatch utility is a tool that allows the application and rollback of interim patches to Oracle products.
	Repository Creation Utility	The Repository Creation Utility (RCU) is used for creating database schemas.

C.2 Downloading Oracle GoldenGate Veridata C-Agent and Java Agent for DB2 for i or DB2 z/OS

Oracle GoldenGate Veridata C-Agent and the Java Agent for DB2 for i or DB2 z/OS platforms are available for download as separate ZIP files on Oracle Technology Network Oracle GoldenGate Downloads page at:

`http://www.oracle.com/technetwork/middleware/goldengate/downloads/index.html`

C.3 Downloading the Product

Follow these steps to download any product from Oracle Software Delivery Cloud:

1. Enter the Oracle Software Delivery Cloud link into a web browser:

   http://edelivery.oracle.com/

2. Click **Sign-in/Register**.

   > **Note:**
   >
   > If you are not already logged in, the Oracle Single Sign-On page appears. Enter your Oracle user id and password and click **Sign In**.

   The Terms & Restrictions page appears.

3. Select the **Oracle Software Delivery Cloud Trial License Agreement** and the **Export Restrictions** check boxes, and then click **Continue**.

   The Media Pack Search page appears.

4. On the Media Pack Search page, do the following:

   a. From the **Select Product Pack** drop-down list, select **Oracle Management Pack for Oracle Goldengate** .

   b. From the **Platform** drop-down list, select the platform on which you are installing Oracle GoldenGate Veridata.

   c. Click **Continue**.

5. The latest release is automatically selected so click **Continue**.

6. Review and accept the terms and restrictions then click **Continue**.

7. Click the individual file names or click **Download All** to begin the download.

8. Extract the ZIP file to a temporary directory.

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

D

Oracle GoldenGate Veridata Silent Installation

To run the Oracle GoldenGate Veridata silent installation, you must provide a valid response file that contains information about the installation.

Prerequisites

For more information, see Prerequisites for Installing Oracle GoldenGate Veridata .

Sample Response File for Oracle GoldenGate Veridata Installation

```
[ENGINE]
#DO NOT CHANGE THIS.
VERSION=1.0.0.0.0
[Generic]
#The Middleware home location.
MIDDLEWARE_HOME=
#The Oracle home location.
ORACLE_HOME=
#Set this to false if you wish to update software automatically
DECLINE_AUTO_UPDATES=false
#If the Software updates are already downloaded and available on your local system, then
specify the path
to the directory where these patches are available and set SPECIFY_DOWNLOAD_LOCATION to
true
AUTO_UPDATES_LOCATION=
#Set this variable value to the Installation Type selected. e.g. SOA Foundation with
Examples,Complete
INSTALL_TYPE=SOA Foundation with Examples
```

Creating Response File

For Oracle GoldenGate Veridata Installer you need a response file saved as follows: `/u01/software/vdt.rsp`:

```
[ENGINE]
VERSION=1.0.0.0.0
[Generic]
MIDDLEWARE_HOME=/scratch/aime1/work/mw1616
ORACLE_HOME=/scratch/aime1/work/mw1616
DECLINE_AUTO_UPDATES=false
AUTO_UPDATES_LOCATION=/net/adcnas418/export/farm_fmwqa/ARU
CUSTOM_TYPE=false
INSTALL_TYPE=SOA Foundation with Examples
[SYSTEM]
[APPLICATIONS]
[RELATIONSHIPS]
```

Oracle GoldenGate Veridata Silent Installation

The following command shows how to initiate the installation in a silent mode:

```
$JAVA_HOME/bin/java -jar /u01/software/fmw_12.2.1.4.0_ogg.jar -silent -
responseFile /u01/software/vdt.rsp.
```

For more information about the parameters in the response files and Silent Installation Syntax and Commands, see Oracle Universal Installer Response File Parameters in *Oracle Fusion Middleware Installing Software with the Oracle Universal Installer* Guide.