Oracle® Fusion Middleware

Installing and Upgrading Oracle GoldenGate for Big Data

12*c* (12.3.0.1)

E80159-01

December 2016

ORACLE®

Oracle Fusion Middleware Installing and Upgrading Oracle GoldenGate for Big Data, 12*c* (12.3.0.1)

E80159-01

Contents

Preface

This preface includes the following topics:

- Audience
- Documentation Accessibility
- Conventions
- Related Information

Audience

This guide is intended for system administrators who are configuring and running Oracle GoldenGate for Big Data.

Documentation Accessibility

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

Access to Oracle Support

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

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.

Convention	Meaning
`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]`.

Related Information

The Oracle GoldenGate Product Documentation Libraries are found at

Oracle GoldenGate

Oracle GoldenGate Application Adapters

Oracle GoldenGate for Big Data

Oracle GoldenGate Director

Oracle GoldenGate Plug-in for EMCC

Oracle GoldenGate Monitor

Oracle GoldenGate for HP NonStop (Guardian)

Oracle GoldenGate Veridata

Oracle GoldenGate Studio

Additional Oracle GoldenGate information, including best practices, articles, and solutions, is found at:

Oracle GoldenGate A-Team Chronicles

1

Installing Oracle GoldenGate for Big Data

This chapter describes how to install a new instance of Oracle GoldenGate for Big Data.

Topics:

Preparing for Installation

Installation Steps

1.1 Preparing for Installation

Prepare your Java environment by ensuring that you have the correct version of Java installed, and that the environmental variables have been set up and configured correctly.

Installation Overview

Downloading Oracle GoldenGate for Big Data

Installing Java

Directory Structure

Setting up Environmental Variables

1.1.1 Installation Overview

This section provides an overview of the installation contents and the Oracle GoldenGate instances used with the Oracle GoldenGate Adapter

Contents of the Installation ZIPFile

Using the Generic Build of Oracle GoldenGate

Considerations for Using a Custom Build for a Big Data Instance of Oracle GoldenGate

Installing to a Non-Generic Instance of Oracle GoldenGate

1.1.1.1 Contents of the Installation ZIPFile

The Oracle GoldenGate for Big Data installation ZIP file contains:

• Oracle GoldenGate Java Adapter

• A version of Oracle GoldenGate designed to stream data to Big Data targets. This version is labeled *generic* because it is not specific to any database, but it is platform dependent.

1.1.1.2 Using the Generic Build of Oracle GoldenGate

For JMS capture, the Java Adapter must run in the generic build of Oracle GoldenGate. However, the generic build is not required when using the adapter for delivery of trail data to a target; in this case, the Java Adapter can be used with any database version of Oracle GoldenGate.

1.1.1.3 Considerations for Using a Custom Build for a Big Data Instance of Oracle GoldenGate

There are both advantages and disadvantages to installing a custom build for a Big Data Oracle GoldenGate instance. Also, there are limitations in the releases of Oracle GoldenGate that are compatible with releases of the Big Data.

Advantages

- The non-generic instance allows you to configure Extract to login to the database for metadata. This removes the need to use a source definitions file that must be synchronized your the source database DDL.

- There is no need to manage two separate versions of Oracle GoldenGate when doing database capture and JMS delivery on the same server.

Disadvantages

- If you need to patch Oracle GoldenGate core instance, you must also copy the Big Data into the new patched installation of Oracle GoldenGate.

- The Oracle GoldenGate for Big Data are only tested and certified with the generic version of Oracle GoldenGate core. New patches of the core can trigger incompatibilities.

Limitations

- The Replicat module to write to Big Data targets is only available in the Generic Oracle GoldenGate distribution.

- The Oracle GoldenGate for Big Data can be installed with the same major release as your Oracle GoldenGate instance. Therefore, 11.1.*x* releases of the Big Data can only be installed to 11.1.*x* releases of Oracle GoldenGate; 11.2.*x* with 11.2.*x*, and 12.1.2.*x* with 12.1.2.*x*.

- The generic build must be used with JMS capture, as this is the only version of Extract that is capable of loading the VAM.

- A `DEFGEN` utility is not included with the Big Data. To generate source definitions, you will need a version of Oracle GoldenGate that is built specifically for your database type.

1.1.1.4 Installing to a Non-Generic Instance of Oracle GoldenGate

If you decide to install the Java user exit to a non-generic instance of Oracle GoldenGate, unzip to a temporary location first and then copy the adapter files to your Oracle GoldenGate installation location

To do this, follow these steps:

1. Extract the Oracle GoldenGate installation ZIP file to a temporary directory.

2. Extract the Oracle GoldenGate installation ZIP file into your Oracle GoldenGate installation directory.

3. Copy or move the files from the temporary directory `ggjava` subdirectory into the Oracle GoldenGate installation directory.

4. Copy or move the shared libraries from the temporary location into the Oracle GoldenGate installation directory.

5. Optionally you can also copy `Gendef`. (There is no need to copy the shared library `ggjava_vam` because it only works with the generic build.)

6. Delete the temporary directory.

1.1.2 Downloading Oracle GoldenGate for Big Data

Oracle GoldenGate for Big Data are available for Windows, Linux, and UNIX. To download, first visit the Oracle support site to see if there is a patch available for your operating system and architecture.

> **Note:**
>
> If you are not planning to use the generic build included in the installation, ensure that the major release of the Oracle GoldenGate for Big Data build you download matches (or is known to be compatible with) the major release of the Oracle GoldenGate instance that will be used with it.

1. Navigate to `http://support.oracle.com`.

2. Sign in with your Oracle ID and password.

3. Select the Patches and Upgrades tab.

4. On the Search tab, click Product or Family.

5. In the Product Field, type **Oracle GoldenGate for Big Data**.

6. From the Release drop-down list, select the release version that you want to download.

7. Make sure Platform is displayed as the default in the next field, and then select the platform from the drop-down list.

8. Leave the last field blank.

9. Click **Search**.

10. In the Advanced Patch Search Results list, select the available builds that satisfy the criteria that you supplied.

11. In the file Download dialog box, click the ZIP file to begin the download.

If patches are not available on the support site, go to the Oracle delivery site for the release download.

1. Navigate to `http://edelivery.oracle.com`.

2. Sign in with your Oracle ID and password.

3. On the Terms and Restrictions page:

- Accept the **Trial License Agreement** (even if you have a permanent license).

- Accept the **Export Restrictions**.

- Click **Continue**.

4. On the Media Pack Search page:

- Select the Oracle Fusion Middleware Product Pack.

- Select the platform on which you will be installing the software.

- Click **Go**.

5. In the Results list:

- Select the Oracle GoldenGate Applications Big Data Media Pack that you want.

- Click **Continue**.

6. On the Download page:

- View the `Readme` file.

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

1.1.3 Installing Java

The Oracle GoldenGate for Big Data are certified for Java 1.8. Before installing and running Oracle GoldenGate for Java, you must install Java (JDK or JRE) version 1.8 or later. Either the Java Runtime Environment (JRE) or the full Java Development Kit (which includes the JRE) may be used.

1.1.4 Directory Structure

The following table is a sample that includes the subdirectories and files that result from unzipping the installation file and creating the subdirectories. The following conventions have been used:

- Subdirectories are enclosed in square brackets []

- Levels are indicated by a pipe and hyphen |‒

- The `Internal` notation indicates a read-only directory that should not be modified

- Text files (`*.txt`) are not included in the list

- Oracle GoldenGate utilities, such as Defgen, Logdump, and Keygen, are not included in the list

Table 1-1   Sample installation directory structure

Directory	Explanation
`[gg_install_dir]`	Oracle GoldenGate installation directory, such as `C:/ggs` on Windows or `/home/user/ggs` on UNIX.
`|-ggsci`	Command line interface used to start, stop, and manage processes.
`|-mgr`	Manager process.
`|-extract`	Extract process that will start the Java application.
`|-replicat`	Replicat process that will start the Java application.
`|-[UserExitExamples]`	Sample C programming language user exit code examples.

`|-[dirprm]`	Subdirectory that holds all the parameter and property files created by the user, for example:
	
	`javaue.prm`
	`javaue.properties`
	`jmsvam.prm`
	`jmsvam.properties`
	`ffwriter.prm`

`|-[dirdef]`	Subdirectory that holds source definitions files (`*.def`) defining the metadata of the trail: • Created by the core utility for the user exit trail data. • Created by the `Gendef` adapter utility for VAM message capture.
`|-[dirdat]`	Subdirectory that holds the trail files produced by the VAM Extract or read by the user exit Extract.
`|-[dirrpt]`	Subdirectory that holds log and report files.
`|-[dirchk]`	`Internal` Subdirectory that holds checkpoint files.
`|-[dirpcs]`	`Internal`Subdirectory that holds process status files.
`|-[dirjar]`	`Internal` Subdirectory that holds Oracle GoldenGate Monitor jar files.
`|-[ggjava]`	`Internal` Installation directory for Java jars. Read-only; do not modify.
`|-|-ggjava.jar`	The main Java application jar that defines the class path and dependencies.

Table 1-1   (Cont.) Sample installation directory structure

Directory	Explanation
`|-|-[resources]`	Subdirectory that contains all `ggjava.jar` dependencies. Includes subdirectories for: • `[class]` - properties and resources • `[lib]` - application jars required by `ggjava.jar`
`|-ggjava_vam.dll`	The VAM shared library. This is `libggjava_vam.so` on UNIX.
`|-ggjava.dll`	Used by the Replicat based delivery process. This is `libggjava.so` on UNIX.
`|-gendef`	Utility to generate the adapter source definitions files containing metadata of the JMS message input (useful only for trail files created by Oracle GoldenGate releases 12.1 or older. Note that this is different from the Oracle GoldenGate `Defgen` utility that creates source definitions containing the input metadata for the trail.
`|-. . .`	Other subdirectories and files included in the installation or created later.

1.1.5 Setting up Environmental Variables

To configure your Java environment for Oracle GoldenGate for Java:

* The `PATH` environmental variable should be configured to find your Java Runtime

* The shared (dynamically linked) Java virtual machine (JVM) library must also be found.

On Windows, these environmental variables should be set as system variables; on Linux/UNIX, they should be set globally or for the user running the Oracle GoldenGate processes. Examples of setting these environmental variables for Windows, UNIX, and Linux are in the following sections.

> **Note:**
>
> There may be two versions of the `JAVA_HOME/.../client`, and another in `JAVA_HOME/.../server`. For improved performance, use the server version, if it is available. On Windows, only the client JVM may be there if only the JRE was installed (and not the JDK).

[Java on Linux/UNIX](#)

[Java on Windows](#)

1.1.5.1 Java on Linux/UNIX

Configure the environment to find the JRE in the `PATH`, and the JVM shared library, using the appropriate environmental variable for your system. For example, on Linux

(and Solaris, etc.), set LD_LIBRARY_PATH to include the directory containing the JVM shared library as follows (for sh/ksh/bash):

Note:

On AIX platforms, you set LIBPATH=. On HP-UX IA64, you set SHLIB_PATH=.

Example 1-1   Configuring path for Java on Linux

```
export JAVA_HOME=/opt/jdk1.8
export PATH=$JAVA_HOME/bin:$PATH
export LD_LIBRARY_PATH=$JAVA_HOME/jre/lib/i386/server:$LD_LIBRARY_PATH
```

In the examples above, the directory $JAVA_HOME/jre/lib/i386/server should contain the libjvm.so and libjsig.so files. The actual directory containing the JVM library depends on the operating system and if the 64-bit JVM is being used.

Verify the environment settings by opening a command prompt and checking the Java version as in this example:

```
$ java -version
java version "1.8.0_92"
Java(TM) SE Runtime Environment (build 1.8.0_92-b14)
```

1.1.5.2 Java on Windows

After Java is installed, configure the PATH to find the JRE and JVM DLL (jvm.dll):

Example 1-2   Configuring Path for Java on Windows

```
set JAVA_HOME=C:\Program Files\Java\jdk1.8.0
set PATH=%JAVA_HOME%\bin;%PATH%
set PATH=%JAVA_HOME%\jre\bin\server;%PATH%
```

In the example above, the directory %JAVA_HOME%\jre\bin\server should contain the file jvm.dll.

Verify the environment settings by opening a command prompt and checking the Java version as in this example:

```
C:\> java -version
java version "1.8.0_92" Java(TM) SE Runtime Environment (build 1.8.0_92-b14))
```

1.2 Installation Steps

Perform the following steps to install the Oracle GoldenGate for Big Data:

1. Create an installation directory that has no spaces in its name. Then extract the ZIP file into this new installation directory. For example:

```
Shell> mkdir installation_directory
Shell> cp path/to/installation_zip installation_directory
Shell> cd installation_directory
Shell> unzip installation_zip
```

If you are on Linux or UNIX, run:

```
Shell> tar -xf installation_tar
```

This downloads the files into several of the subdirectories Directory Structure .

2. Stay on the installation directory and bring up GGSCI to create the remaining subdirectories in the installation location.

```
Shell> ggsci
GGSCI> CREATE SUBDIRS
```

3. Create a Manager parameter file:

```
GGSCI> EDIT PARAM MGR
```

4. Specify a port for the Manager to listen on by using the editor to add a line to the Manager parameter file. For example:

```
PORT 7801
```

5. If you are on Windows and running Manager as a service, set the system variable PATH to include jvm.dll, then delete the Manager service and re-add it.

6. Go to GGSCI, start the Manager, and check to see that it is running:

```
GGSCI>START MGR
GGSCI>INFO MGR
```

> **Note:**
>
> To check for environmental variable problems locating the JVM at runtime:
>
> • Add the parameter GETENV(PATH) for Windows or GETENV(LD_LIBRARY_PATH) for UNIX to the Replicat parameter file.
>
> • Start the Replicat process
>
> • Check the output for the report using the GGSCI command: SEND REPLICAT *group_name* REPORT

2

Upgrading Oracle GoldenGate for Big Data

This chapter describes how to upgrade to Oracle GoldenGate for Big Data 12*c* (12.3.0.1.0) by downloading the product as described in Downloading Oracle GoldenGate for Big Data, and then choosing the upgrade paths that suits your environment.

After upgrading, you must then convert your Extract processes to Replicat processes.

Topics:

Upgrading by Overwriting the Existing Installation

Upgrading by Installing into a New Directory

Switching Existing Extract Processes to Replicat Processes

2.1 Upgrading by Overwriting the Existing Installation

The most straightforward upgrade path is to copy the Oracle GoldenGate for Big Data 12*c* (12.3.0.*x*) files into the existing 12*c* (12.2.0.*x*) installation directory. Overwriting the product files is possible because there is neither structural nor package name changes in 12.3.0.*x*. The 12.2.0.*x* handler and formatter configurations are also compatible with the 12.3.0.*x* release.

1. (Source and target systems) Back up the current Oracle GoldenGate for Big Data installation directory on the source and target systems, and any working directories that you have installed on a shared drive in a cluster (if applicable).

2. (Source and target systems, as applicable) Expand version 12*c* (12.2.0.1) of Oracle GoldenGate into a new directory on each system (not the current Oracle GoldenGate directory). Do not create the sub-directories; just complete the steps to the point where the installation files are expanded.

3. (Source system) Stop user activity on objects in the Oracle GoldenGate configuration.

4. Create an installation directory with *no* spaces in its name.

5. Extract the ZIP file into this new installation directory, which divides the files into several subdirectories.

6. Copy the Oracle GoldenGate for Big Data 12*c* (12.3.0.*x*) files into the existing 12*c* (12.2.0.*x*) installation directory.

7. Start the Replicat processes and verify that they are running.

   ```
   GGSCI> START MANAGER
   GGSCI> START REPLICAT group_name
   GGSCI> INFO REPLICAT group_name
   GGSCI> VIEW REPORT group_name
   ```

Certain 12.2.0.*x* JAR files are still present after the overwriting process though the new 12.3.0.1.*x* JAR files are used.

Related Topics:

Switching Existing Extract Processes to Replicat Processes

2.2 Upgrading by Installing into a New Directory

Use the following steps to install the Oracle GoldenGate for Big Data 12*c* (12.3.0.*x*) files into a new installation directory.

1. (Source and target systems) Back up the current Oracle GoldenGate for Big Data installation directory on the source and target systems, and any working directories that you have installed on a shared drive in a cluster (if applicable).

2. (Source and target systems, as applicable) Expand version 12*c* (12.2.0.1) of Oracle GoldenGate into a new directory on each system (not the current Oracle GoldenGate directory). Do not create the sub-directories; just complete the steps to the point where the installation files are expanded.

3. (Source system) Stop user activity on objects in the Oracle GoldenGate configuration.

4. Create an installation directory with *no* spaces in its name.

5. Extract the ZIP file into this new installation directory, which divides the files into several subdirectories.

6. In the installation directory, start GGSCI to create the remaining subdirectories in the installation location.

   ```
   $ ggsci
   GGSCI> CREATE SUBDIRS
   ```

7. Copy all of the `dirprm` files from your existing installation into the `dirprm` directory in the new installation location.

   > **Note:**
   >
   > All of your configuration files must be in the `dirprm` directory. If you have property files, Velocity templates, or other configuration files in a location other than `dirprm` in your old installation, then you must copy them to the `dirprm` directory in the new installation.

8. Copy all of the `dirdef` files from your existing installation into the `dirdef` directory in the new installation location.

9. If you have data files stored in the 12.2.0.*x* installation `dirdat` directory, then copy or move the existing trail files to the `dirdat` directory of the new installation.

10. If you have additional JAR files or other custom files in your 12.2.0.*x* installation, then copy them to the new installation directory.

11. Configure the Replicat processes in the new installation directory by starting GGSCI and adding the Replicat and naming the trails.

```
GGSCI> ADD REPLICAT group_name, EXTTRAIL trail_name, ...
GGSCI> ALTER group_name EXTSEQNO seqno EXTRBA rba
```

Optionally, you could alter the starting position of Replicat processing as needed.

12. Start the Replicat processes and verify that they are running.

```
GGSCI> START MANAGER
GGSCI> START REPLICAT group_name
GGSCI> INFO REPLICAT group_name
GGSCI> VIEW REPORT group_name
```

13. Modify the source system to write to the new Oracle GoldenGate for Big Data installation directory:

   a. (Optional) Upgrade the source database Oracle GoldenGate capture following the upgrade procedure for your database platform.

   b. Configure the source database capture to write to the new Oracle GoldenGate for Big Data 12.3.0.*x* installation `dirdat` directory.

   c. When the old Oracle GoldenGate for Big Data installation has processed all its data, switch over to the process that will send data to the new location.

Related Topics:

2.3 Switching Existing Extract Processes to Replicat Processes

In previous releases, you could use an Extract and pump process to write to your Big Data targets. In this release, this solution is deprecated so you must use Replicat.

A typical Extract configuration is similar to:

```
EXTRACT mygroup
SOURCEDEFS path/to/source/def/file
CUSEREXIT libggjava_ue.so CUSEREXIT PASSTHRU INCLUDEUPDATEBEFORES
GETUPDATEBEFORES
TABLE *.*;
```

With Replicat the preceding configuration would be:

```
REPLICAT mygroup
SOURCEDEFS path/to/source/def/file
TARGETDB LIBFILE libggjava.so SET property=/path/to/properties/file
MAP *.* TARGET *.*;
```

The same properties file works in the Replicat configuration so you do not need to change the properties file.

To complete this process, you must add the Replicat group, start the process, and verify that it is running:

```
GGSCI> ADD REPLICAT group_name, EXTTRAIL trail_name, ...
GGSCI> ALTER group_name EXTSEQNO seqno EXTRBA rbaGGSCI> START REPLICAT group_name
GGSCI> INFO REPLICAT group_name
GGSCI> VIEW REPORT group_name
```