

**Oracle® GoldenGate for DB2 for i**

Installation and Setup Guide

Release 11.2.1.0.2

**E29643-01**

August 2012

Copyright © 1995, 2012, Oracle and/or its affiliates. All rights reserved.

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

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

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

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

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

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

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

---

---

# Contents

<b>Preface</b> .....	vii
<b>1 System requirements and preinstallation instructions</b>	
<b>Overview of Oracle GoldenGate for DB2 for i</b> .....	1-1
<b>Supported Platforms</b> .....	1-1
<b>Operating system requirements</b> .....	1-1
General requirements .....	1-2
Memory requirements .....	1-2
Disk requirements .....	1-2
Network requirements .....	1-3
Oracle GoldenGate security privileges .....	1-3
<b>Supported DB2 for i data types</b> .....	1-4
<b>Non-supported DB2 for i data types</b> .....	1-5
<b>Supported objects and operations for DB2 for i</b> .....	1-5
<b>Non-supported objects and operations for DB2 for i</b> .....	1-5
<b>Supported object naming conventions</b> .....	1-5
<b>2 Installing Oracle GoldenGate</b>	
<b>Installation overview</b> .....	2-1
<b>Downloading Oracle GoldenGate</b> .....	2-1
<b>Setting library paths for dynamic builds on Linux</b> .....	2-2
<b>Preparing to install Oracle GoldenGate within a cluster</b> .....	2-3
<b>Installing Oracle GoldenGate on IBM i</b> .....	2-4
<b>Installing Oracle GoldenGate on Linux</b> .....	2-4
<b>Installing Oracle GoldenGate on Windows</b> .....	2-5
Installing Oracle GoldenGate into a Windows Cluster .....	2-5
Installing the Oracle GoldenGate files .....	2-5
Specifying a custom Manager name .....	2-5
Installing Manager as a Windows service .....	2-6
<b>Integrating Oracle GoldenGate into a cluster</b> .....	2-7
General requirements in a cluster .....	2-7
Adding Oracle GoldenGate as a Windows cluster resource .....	2-8
<b>3 Preparing the system for Oracle GoldenGate</b>	
<b>Preparing the journals for data capture by Extract</b> .....	3-1

Allocating journals to an Extract group .....	3-1
Setting journal parameters .....	3-1
Deleting old journal receivers .....	3-2
<b>Specifying object names .....</b>	<b>3-2</b>
<b>Preparing tables for processing .....</b>	<b>3-3</b>
Assigning row identifiers .....	3-3
Preventing key changes .....	3-4
Disabling constraints on the target .....	3-4
Enabling change capture .....	3-4
Maintaining materialized query tables .....	3-5
Specifying the Oracle GoldenGate library .....	3-6
<b>Adjusting the system clock .....</b>	<b>3-6</b>
<b>Configuring the ODBC driver .....</b>	<b>3-6</b>
Configuring ODBC on Linux .....	3-6
Configuring ODBC on Windows .....	3-10

## **4 Configuring Oracle GoldenGate on source and target databases**

<b>What to expect from this procedure.....</b>	<b>4-1</b>
<b>Creating and editing parameter files.....</b>	<b>4-1</b>
<b>Overview of basic steps to configure Oracle GoldenGate.....</b>	<b>4-2</b>
<b>Choosing names for processes and files.....</b>	<b>4-2</b>
Choosing group names .....	4-2
Choosing trail file names .....	4-3
<b>Creating the Oracle GoldenGate instance.....</b>	<b>4-3</b>
<b>Creating a data definitions file.....</b>	<b>4-4</b>
<b>Configuring Extract for change capture.....</b>	<b>4-5</b>
Encrypting the Extract passwords .....	4-5
Configuring the primary Extract .....	4-5
Configuring the data pump .....	4-6
<b>Configuring Replicat for change delivery.....</b>	<b>4-8</b>
Creating a checkpoint table .....	4-8
Encrypting the Replicat password .....	4-9
Configuring Replicat .....	4-9
<b>Configuring additional process groups for best performance.....</b>	<b>4-10</b>
<b>Oracle GoldenGate parameters not supported for DB2 for i.....</b>	<b>4-11</b>
<b>Next steps in the deployment.....</b>	<b>4-11</b>
<b>When to start replicating transactional changes.....</b>	<b>4-12</b>
<b>Testing your configuration.....</b>	<b>4-12</b>

## **5 Instantiating and starting Oracle GoldenGate replication**

<b>About the instantiation process.....</b>	<b>5-1</b>
<b>Overview of basic Oracle GoldenGate instantiation steps.....</b>	<b>5-1</b>
<b>Satisfying prerequisites for instantiation.....</b>	<b>5-2</b>
Configure change capture and delivery .....	5-2
Add collision handling .....	5-2
Prepare the target tables .....	5-2
<b>Making the instantiation procedure more efficient.....</b>	<b>5-3</b>

Share parameters between process groups .....	5-3
Use parallel processes .....	5-3
<b>Configuring the initial load .....</b>	<b>5-3</b>
Configuring an initial load from file to Replicat .....	5-3
Configuring an initial load with a database utility .....	5-7
<b>Adding change-capture and change-delivery processes.....</b>	<b>5-7</b>
Add the primary Extract .....	5-7
Add the local trail .....	5-9
Add the data pump Extract group .....	5-9
Add the remote trail .....	5-10
Add the Replicat group .....	5-10
<b>Performing the target instantiation .....</b>	<b>5-10</b>
To perform instantiation from file to Replicat .....	5-11
To perform instantiation with a database utility .....	5-12
<b>Monitoring processing after the instantiation.....</b>	<b>5-13</b>
<b>Backing up the Oracle GoldenGate environment .....</b>	<b>5-13</b>
<b>Positioning Extract after startup .....</b>	<b>5-14</b>

## **6 Uninstalling Oracle GoldenGate**

Uninstalling Oracle GoldenGate from a source DB2 for i system .....	6-1
Uninstalling Oracle GoldenGate from a Linux remote system .....	6-2
<b>Removing Oracle GoldenGate from a Windows remote system.....</b>	<b>6-2</b>
Removing Oracle GoldenGate from a Windows cluster .....	6-2
Removing Oracle GoldenGate from a remote Windows system .....	6-3

## **A Oracle GoldenGate installed components**

Oracle Goldengate Programs And Utilities .....	A-1
Oracle Goldengate Subdirectories .....	A-2
Other Oracle GoldenGate files.....	A-4
Oracle GoldenGate checkpoint table .....	A-8



---

---

# Preface

This guide helps you get started with installing, configuring, and running Oracle GoldenGate on a DB2 for i database. With Oracle GoldenGate for DB2 for i, you can:

- Map, filter, and transform transactional data changes between similar or dissimilar supported DB2 for i versions, and between supported DB2 for i versions and other supported types of databases.
- Perform initial loads from DB2 for i to target tables in DB2 for i or other databases to instantiate a synchronized replication environment.

This documentation is meant to be a step by step guide in establishing a basic configuration of Oracle GoldenGate. It should be followed in a linear fashion, so that you benefit from important information in previous sections. It is not meant to be used as a reference document.

## Audience

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

## 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 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 Documents

The complete Oracle GoldenGate documentation set includes the following components:

### HP NonStop Platform

- *Oracle GoldenGate for NonStop Reference Guide*

- *Oracle GoldenGate for NonStop Administrator's Guide*

### **Windows, UNIX, and Linux Platforms**

- *Oracle GoldenGate Installation and Setup Guides per supported database*
- *Oracle GoldenGate Windows and UNIX Administrator's Guide*
- *Oracle GoldenGate Windows and UNIX Reference Guide*
- *Oracle GoldenGate Windows and UNIX Troubleshooting and Tuning Guide*
- *Oracle GoldenGate Upgrade Guide*

### **Other Oracle GoldenGate Products**

- *Oracle GoldenGate Adapter for Flat Files Administrator's Guide*
- *Oracle GoldenGate Adapter for Java Administrator's Guide*
- *Oracle GoldenGate Director Administrator's Guide*
- *Oracle GoldenGate Monitor Administrator's Guide*
- *Oracle GoldenGate Veridata Administrator's Guide*

## **Conventions**

The following text conventions are used in this document:

<b>Convention</b>	<b>Meaning</b>
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, such as "From the File menu, select <b>Save</b> ." Boldface also is used for terms defined in text or in the glossary.
<i>italic</i> <i>italic</i>	Italic type indicates placeholder variables for which you supply particular values, such as in the parameter statement: <code>TABLE <i>table_name</i></code> . 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: <code>{<i>option1</i>   <i>option2</i>   <i>option3</i>}</code> .
[ ]	Brackets within syntax indicate an optional element. For example in this syntax, the <code>SAVE</code> clause is optional: <code>CLEANUP REPLICAT <i>group_name</i> [, <i>SAVE count</i>]</code> . Multiple options within an optional element are separated by a pipe symbol, for example: <code>[<i>option1</i>   <i>option2</i>]</code> .

---

---

# System requirements and preinstallation instructions

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

## 1.1 Overview of Oracle GoldenGate for DB2 for i

With Oracle GoldenGate for DB2 for i, you can replicate data to and from similar or dissimilar supported DB2 for i versions, or you can move data between a DB2 for i database and a database of another type. Oracle GoldenGate for DB2 for i supports the filtering, mapping, and transformation of data unless otherwise noted in this documentation.

Oracle GoldenGate for DB2 for i runs directly on a DB2 for i source system to capture data from the transaction journals for replication to a target system. To apply data to a DB2 for i database, Oracle GoldenGate runs on a remote Windows or Linux system and delivers the data by means of a remote ODBC connection. No Oracle GoldenGate software is installed on a DB2 for i target.

---

---

**Note:** The IBM i platform uses one or more *journals* to keep a record of transaction change data. For consistency of terminology in the supporting administrative and reference Oracle GoldenGate documentation, the terms "log" or "transaction log" may be used interchangeably with the term "journal" where the use of the term "journal" is not explicitly required.

---

---

## 1.2 Supported Platforms

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

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

## 1.3 Operating system requirements

This section outlines the requirements to support Oracle GoldenGate on the local operating system.

### 1.3.1 General requirements

- Portable Application Solution Environment (PASE) must be installed on the system.
- OpenSSH is recommended to be installed on the system. OpenSSH is part of the IBM Portable Utilities licensed program and allows SSH terminal access to the system in the same manner as other Linux system.

### 1.3.2 Memory requirements

The amount of memory that is required for Oracle GoldenGate depends on the number of concurrent processes that will be running. The following are the minimum number of processes that are required on an IBM i system (you may find you need more during testing):

- A primary Extract process that captures source data
- A secondary Extract data-pump process that transfers data across the network

---

---

**Note:** Oracle GoldenGate also requires a Replicat process to deliver data to the target system. No Oracle GoldenGate software is installed on a DB2 for i system. Replicat runs on a Windows or Linux system and delivers data to an IBM i target through remote ODBC. See XREF for more information.

---

---

On a source DB2 for i system, Oracle GoldenGate supports up to 5,000 concurrent Extract processes per instance of Manager (the Oracle GoldenGate controller process). Each Extract and Replicat process needs approximately 25-55 MB of memory, or more depending on the number of journals that Extract will be processing, the size of the transactions, and the number of concurrent transactions.

The amount of physical memory that is used by Oracle GoldenGate is controlled by the operating system, not the Oracle GoldenGate processes. The Oracle GoldenGate cache manager takes advantage of the memory management functions of the operating system to ensure that the Oracle GoldenGate processes work in a sustained and efficient manner.

To provide enough virtual memory to the Oracle GoldenGate cache manager, the recommended setting for the PASE MAXSHR64 environment variable is a value of 512 (128GB) or higher. Set this variable before running any Oracle GoldenGate processes. For more information about evaluating Oracle GoldenGate memory requirements, see the CACHEMGR parameter in the *Oracle GoldenGate for Windows and UNIX Reference Guide*.

---

---

**Note:** If MAXSHR64 is not set, you may encounter a warning message stating that the virtual memory is less than the recommended amount. Unless you have very large long-running transactions or a very large number of concurrent transactions, you can ignore this message.

---

---

### 1.3.3 Disk requirements

Assign the following free disk space:

- 50-150 MB, depending on the database and platform. This includes space for the compressed download file and space for the uncompressed files. You can delete the download file after the installation is complete.

- 40 MB for the working directories and binaries for each instance of Oracle GoldenGate that you are installing on the system. For example, to install two builds of Oracle GoldenGate into two separate directories, allocate 80 MB of space.
- An additional 1 GB of disk space on any system that hosts Oracle GoldenGate trails, which are files that contain the working data. You may need more or less than this amount, because the space that is consumed by the trails depends on the volume of data that will be processed. See the guidelines for sizing trails in the *Oracle GoldenGate for Windows and UNIX Administrator's Guide*.

### 1.3.4 Network requirements

Oracle GoldenGate requires the following network resources.

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

### 1.3.5 Oracle GoldenGate security privileges

This section outlines the security privileges that Oracle GoldenGate requires on a source DB2 for i system and on a Windows or Linux target system.

---

---

**Note:** In each case you will be assigning a user account. Keep a record of each database user. These users must be specified in the Oracle GoldenGate parameter files.

---

---

#### 1.3.5.1 Oracle GoldenGate security privileges on an IBM i source system

On an IBM i source system, Manager and the Extract process are active. The DEFGEN utility also may be used if you are replicating data to a dissimilar target system.

The person who installs Oracle GoldenGate must have read and write privileges on the Oracle GoldenGate installation directory, because steps will be performed to create some sub-folders and run some programs.

The Oracle GoldenGate processes must be assigned a user profile account that is dedicated to Oracle GoldenGate and cannot be used by any other program. One user profile can be used by all of the Oracle GoldenGate processes. This profile need only be granted permission to the objects that Oracle GoldenGate will be operating upon. If specific change data is not to be seen by Oracle GoldenGate, do not include it in any of the journals that the Oracle GoldenGate user profile is allowed to access. All Oracle

GoldenGate processes must have privileges to read, write, and delete files and directories within the Oracle GoldenGate installation directory.

The Manager process must have privileges to control all other Oracle GoldenGate processes (IBM i \*JOBCTL authority).

Assign \*USE authority to all objects on the system that the Extract user profile must have access to. This can be accomplished by either granting \*ALLOBJ authority to the Extract user, or by setting the individual authority to the objects (FILE, LIBRARY and JOURNAL objects) that the Extract user must access. This includes the objects in the QSYS2 library where the SQL catalog resides. These authorities must be granted through the native IBM i interface through a 5250 terminal session or through the IBM i Operations Navigator product available from IBM.

### 1.3.5.2 Oracle GoldenGate security privileges on a Windows or Linux system

On a Windows or Linux system, Manager, Replicat, and Collector (program name is "server") are active. Manager controls the other processes and interacts with Collector to receive incoming data, while Replicat applies data to the target DB2 for i database through ODBC.

The person who installs Oracle GoldenGate must have read and write privileges on the Oracle GoldenGate installation directory, because steps will be performed to create some sub-folders and run some programs. On Windows, the person who installs Oracle GoldenGate must log in as Administrator.

Oracle GoldenGate processes must be assigned a user account that is dedicated to Oracle GoldenGate and cannot be used by any other program. One user account can be used by all of the Oracle GoldenGate processes. This account must have privileges to read, write, and delete files and directories within the Oracle GoldenGate installation directory.

If the Extract user profile does not have the required authority, Extract will log the following errors and stop.

```
[SC=-1224:SQL1224N A database agent could not be started to service a request, or was terminated as a result of a database system shutdown or a force command. SQL STATE 55032: The CONNECT statement is invalid, because the database manager was stopped after this application was started]
```

## 1.4 Supported DB2 for i data types

Oracle GoldenGate supports all DB2 for i data types, except those listed in [Non-supported DB2 for i data types](#).

### Limitations of support

- Extract fully supports the capture and apply of `TIMESTAMP(0)` through `TIMESTAMP(6)`. Extract also captures `TIMESTAMP(7)` through `TIMESTAMP(12)`, but it truncates the data to microseconds (maximum of six digits of fractional time) and issues a warning to the error log. Replicat truncates timestamp data from other sources to microseconds when applying it to `TIMESTAMP(7)` through `TIMESTAMP(12)` in a DB2 for i target.
- When the size of a large object exceeds 4K, Oracle GoldenGate stores the data in segments within the Oracle GoldenGate trail. The first 4K is stored in the base segment, and the rest is stored in a series of 2K segments. Oracle GoldenGate does not support the filtering, column mapping, or manipulation of large objects of this size. Full Oracle GoldenGate functionality can be used for objects that are 4K or smaller.

## 1.5 Non-supported DB2 for i data types

Oracle GoldenGate does not support the following IBM for i data types.

- XML
- DATALINK
- DECFLOAT
- User-defined types

## 1.6 Supported objects and operations for DB2 for i

Oracle GoldenGate supports the following DB2 for i objects and operations.

- Only the default DB 2 for i database that is identified by the system name (in upper case) is supported. Named databases on other independent auxiliary storage pools (IASPs) are not supported
- Extraction and replication of DML operations
- Tables that contain rows of up to 32 KB in length.
- Tables that contain up to the maximum number of columns that is supported by the database, up to the maximum supported column size.
- TRUNCATE TABLE and Clear Physical File Member (CLRPFM)
- Base tables underlying Materialized Query Tables, but not the MQTs themselves. The target database automatically maintains the content of the MQT based on the changes that are applied to the base tables by Replicat.
- Aliases (to allow replication of members other than the first member in a \*FILE object when using SQL tables)

## 1.7 Non-supported objects and operations for DB2 for i

Oracle GoldenGate does not support the following objects or operations for DB2 for i.

- DDL operations
- Schema, table or column names that have trailing spaces.
- Multiple instances of a database
- Partitioned tables

## 1.8 Supported object naming conventions

Oracle GoldenGate supports SQL naming conventions and also supports native file system names in the format of *library/file(member)*.

For native (system) names, Oracle GoldenGate supports the normal IBM i naming rules for wildcarding, which allows \*ALL or a partial name with a trailing asterisk (\*) wildcard. For example:

- *library/\*all(\*all)*
- *library/a\*(a\*)*
- *library/abcde\**

Oracle GoldenGate does not allow wildcarding for library names.

The member name is optional and may be left off. In that case, data for all of the members will be extracted, but only the library and file names will be captured and included in the records that are written to the trail. The result is that the data will appear to have come from only one member on the source, and you should be aware that this could cause integrity conflicts on the target if there are duplicate keys across members. To include the member name in the trail records, include the member explicitly or through a wildcarded member specification.

For SQL names, only the first member in the underlying native file is extracted in accordance with the normal operation of SQL on an IBM i system. For SQL names, Oracle GoldenGate supports the wildcarding of table names, but not schema names. For instructions on wildcarding SQL names, see the *Oracle GoldenGate for Windows and UNIX Administrator's Guide*.

---

---

# Installing Oracle GoldenGate

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

<http://www.oracle.com/technology/software/products/goldengate/index.html>

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

The installation process takes a short amount of time.

## 2.1 Installation overview

Oracle GoldenGate for DB2 for i is installed directly on a source IBM i system, but not on an IBM i target system. To deliver to an IBM i target, Oracle GoldenGate is installed on a Windows or Linux system, and Replicat connects to the target system through ODBC. To install Oracle GoldenGate, the following steps are required:

[Downloading Oracle GoldenGate](#)

[Setting library paths for dynamic builds on Linux](#)

[Preparing to install Oracle GoldenGate within a cluster](#)

[Installing Oracle GoldenGate on Linux](#)

[Installing Oracle GoldenGate on Windows](#)

[Integrating Oracle GoldenGate into a cluster](#)

## 2.2 Downloading Oracle GoldenGate

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

1. Navigate to <http://edelivery.oracle.com>
2. On the Welcome page:
  - Select your language.
  - Click **Continue**.
3. On the Export Validation page:
  - Enter your identification information.
  - Accept the Trial License Agreement (even if you have a permanent license).

- Accept the Export Restrictions.
  - Click **Continue**.
4. On the Media Pack Search page:
    - Select the Oracle Fusion Middleware Product Pack.
    - Select the platform on which you will be installing the software.
    - Click **Go**.
  5. In the Results List:
    - Select the Media Pack that you want to download.

---



---

**Note:** To install Oracle GoldenGate to connect to a DB2 for i target, select the appropriate build for the target IBM i operating system, not the Windows or Linux system on which you actually will be installing Oracle GoldenGate.

---



---

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

---



---

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

---



---

## 2.3 Setting library paths for dynamic builds on Linux

Oracle GoldenGate uses shared libraries. When you install Oracle GoldenGate on an IBM i system, the following must be true *before you run GGSCI or any other Oracle GoldenGate process*.

If you will be running an Oracle GoldenGate program from outside the Oracle GoldenGate installation directory on an IBM i system, do the following:

- (Optional) Add the Oracle GoldenGate installation directory to the `PATH` environment variable.
- (Required) Add the Oracle GoldenGate installation directory to the `shared-libraries` environment variable.

For example, given an Oracle GoldenGate installation directory of `/users/ogg`, the second command in the following example requires these variables to be set:

Command	Requires GG libraries in environment variable?
<code>\$ users/ogg &gt; ./ggsci</code>	No
<code>\$ users &gt; ./ogg/ggsci</code>	Yes

### To set the variables in Korn shell

```
PATH=installation_directory:$PATH
export PATH
LIBPATH=absolute_path_of_installation_directory:$LIBPATH
```

```
export shared_libraries_variable
```

#### To set the variables in Bourne shell

```
export PATH=installation_directory:$PATH
export LIBPATH=absolute_path_of_installation_directory:$LIBPATH
```

#### To set the variables in C shell

```
setenv PATH installation_directory:$PATH
setenv LIBPATH absolute_path_of_installation_directory:$LIBPATH
```

---

**Note:** To view the libraries that are required by an Oracle GoldenGate process, use the `ldd process` shell command before starting the process. This command also shows an error message for any that are missing.

---

## 2.4 Preparing to install Oracle GoldenGate within a cluster

Oracle GoldenGate can be used with any cluster-management solution that is Oracle-certified. You will need to install at least some Oracle GoldenGate objects on shared storage. Select cluster-aware shared storage that is independent of, but available to, all nodes of the cluster.

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

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

- The Oracle GoldenGate installation must have the same location path on every node
- At minimum, install the following directories on the shared storage to support Oracle GoldenGate recovery requirements. On IBM i, you can create symbolic links to them from the installation directory on each node.
  - br
  - dirchk
  - dirdat
  - dirtmp

These directories are among those created when you issue `CREATE SUBDIRS` during installation.

- The parameter files in the `dirprm` directory, if not placed on the shared drive, must be identical on all nodes. To resolve environment settings that must be different from one node to the other, you can set environment settings so they are inherited from the local Manager process or reference a node-specific Oracle GoldenGate macro file. Because this scenario can be difficult to enforce, the inherent concerns can be avoided by storing the parameter files on the shared drive.

See also ["Integrating Oracle GoldenGate into a cluster"](#) on page 2-7 after you install Oracle GoldenGate.

## 2.5 Installing Oracle GoldenGate on IBM i

Follow these steps to install Oracle GoldenGate on an IBM i system.

1. Extract the Oracle GoldenGate `mediapack.zip` file to the system and directory where you want Oracle GoldenGate to be installed.
2. Run the command shell.
3. Change directories to the new Oracle GoldenGate directory.
4. From the Oracle GoldenGate directory, run the GGSCI program.
5. In GGSCI, issue the following command to create the Oracle GoldenGate working directories.

```
CREATE SUBDIRS
```

6. Issue the following command to exit GGSCI.

```
EXIT
```

7. In the Oracle GoldenGate directory, run the shell script `ggos400install`. There is an option parameter for `ggos400install` that specifies the native library that the journal reader native object is installed into. If the library is not provided, the current library of the user running `ggos400install` will be used. At the completion of `ggos400install`, a symbolic link to the native journal reader object is created that is used by the Extract and GGSCI programs to determine where to locate the native object `OGGPRCJRN`.

---

---

**Note:** Do not delete this symbolic link, or Oracle GoldenGate programs will not be able to run. If the symbolic link to `OGGPRCJRN` is deleted by mistake, re-run `ggos400install` again using the same library argument to recreate the link.

---

---

## 2.6 Installing Oracle GoldenGate on Linux

Follow these steps to install Oracle GoldenGate for Oracle on a Linux system.

1. Extract the Oracle GoldenGate `mediapack.zip` file to the system and directory where you want Oracle GoldenGate to be installed.
2. Run the command shell.
3. Change directories to the new Oracle GoldenGate directory.
4. From the Oracle GoldenGate directory, run the GGSCI program.

```
GGSCI
```

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

```
CREATE SUBDIRS
```

6. Issue the following command to exit GGSCI.

```
EXIT
```

## 2.7 Installing Oracle GoldenGate on Windows

Follow these steps to install Oracle GoldenGate for Oracle on a Windows system.

### 2.7.1 Installing Oracle GoldenGate into a Windows Cluster

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

### 2.7.2 Installing the Oracle GoldenGate files

1. Unzip the downloaded file(s) by using WinZip or an equivalent compression product.
2. Move the files in binary mode to a folder on the drive where you want to install Oracle GoldenGate. *Do not* install Oracle GoldenGate into a folder that contains spaces in its name, even if the path is in quotes. For example:

```
C:\ "Oracle GoldenGate" is not valid.
```

```
C:\Oracle_GoldenGate is valid.
```

3. From the Oracle GoldenGate folder, run the GGSCI program.
4. In GGSCI, issue the following command to create the Oracle GoldenGate working directories.

```
CREATE SUBDIRS
```

5. Issue the following command to exit GGSCI.

```
EXIT
```

### 2.7.3 Specifying a custom Manager name

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

- You want to use a name for Manager other than the default of GGSMGR.
- There will be multiple Manager processes running as Windows services on this system. Each Manager on a system must have a unique name. Before proceeding further, note the names of any local Manager services.

#### To specify a custom Manager name

1. From the directory that contains the Manager program, run GGSCI.
2. Issue the following command.

```
EDIT PARAMS ./GLOBALS
```

---

**Note:** The ./ portion of this command must be used, because the GLOBALS file must reside at the root of the Oracle GoldenGate installation file.

---

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

```
MGRSERVNAME name
```

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

## 2.7.4 Installing Manager as a Windows service

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

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

### To install Manager as a Windows service

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

```
install option [...]
```

Where *option* is one of the following:

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

Option	Description
PASSWORD <i>password</i>	Specifies the password for the user that is specified with USER.

<sup>1</sup> A user account can be changed by selecting the Properties action from the Services applet of the Windows Control Panel.

- (Windows Server 2008) If Windows User Account Control (UAC) is enabled, you are prompted to allow or deny the program access to the computer. Select **Allow** to enable the INSTALL utility to run.

The INSTALL utility installs the Manager service with a local system account running with administrator privileges. No further UAC prompts will be encountered when running Manager if installed as a service.

---

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

---

## 2.8 Integrating Oracle GoldenGate into a cluster

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

### 2.8.1 General requirements in a cluster

- Register the Oracle GoldenGate Manager process (and only Manager) as a cluster-managed resource as you would any other application. Manager must be the only Oracle GoldenGate process that the cluster-management software starts and stops, because it is the parent process that manages all other processes.
- If the cluster uses a virtual IP address, you may need to obtain an available fixed IP address for the Manager process. The VIP must be an available IP address on the public subnet and cannot be determined through DHCP. In the parameter files of the Extract data pumps, specify the VIP of the remote Manager as the input value of the `RMTHOST` parameter. Other Oracle GoldenGate products that access Manager also should use the VIP.
- Make certain that all nodes in the cluster have synchronized system clocks. The clocks must be synchronized with the clock on the system where Extract is executed. Oracle GoldenGate compares the time of the local system to the commit timestamps to make critical decisions. For information about synchronizing system clocks, consult <http://www.ntp.org> or your systems administrator.
- When you configure Manager, add the `AUTOSTART` and `AUTORESTART` parameters so that Manager starts the replication processes automatically. You can, when needed, control Extract, Replicat, and other Oracle GoldenGate processes from within the Oracle GoldenGate user interfaces.
- Mount the shared drive on one node only. This prevents processes from being started on another node. Use the same mount point on all nodes.
- Configure Oracle GoldenGate as directed in this documentation.

## 2.8.2 Adding Oracle GoldenGate as a Windows cluster resource

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

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

---

---

## Preparing the system for Oracle GoldenGate

This chapter contains guidelines for preparing the system to support Oracle GoldenGate.

### 3.1 Preparing the journals for data capture by Extract

All tables for which you want data to be captured must be journaled, either explicitly or by default by means of a QSQJRN journal in the same library. To preserve data integrity, data journal entries are sent to Extract in time order as they appear on the system. This section provides guidelines for configuring the journals to support capture by Extract.

#### 3.1.1 Allocating journals to an Extract group

One Extract process can handle up to 30 journals. If using more journals than that, use additional Extract processes to handle the extra journals. You can also use additional Extract processes to improve capture performance if necessary.

---

---

**Note:** To ensure transaction integrity, all journals that correspond to any given transaction must be read by the same Extract group. For more information about using multiple Extract processes, see the *Oracle GoldenGate for Windows and UNIX Administrator's Guide*.

---

---

#### 3.1.2 Setting journal parameters

To support the capture of data by the Extract process, the following journaling parameter settings are required.

- Manage Receivers (MNGRCV) : \*SYSTEM
- Delete Receivers (DLTRCV) : \*NO
- Receiver Size Option (RCVSIZEOPT) : \*MAXOPT2 (\*MAXOPT3 recommended)
- Journal State (JRNSTATE) : \*ACTIVE
- Minimize Entry Specific Data (MINENTDTA) : \*NONE
- Fixed Length Data (FIXLENDTA) : \*USR \*SYSSEQ

---

---

**Note:** You can use `MINENTDTA(*FLDBDY)`, but this setting does not journal all columns, and the `FETCHCOLS` parameter must be used in the Extract configuration to get the data for the missing columns. Note that in this case there will be no way to ensure that the fetched data is accurate relative to the journal entry at the time that the journal entry is read. Never use the option `MINENTDTA(*FILE)` for a journal that Oracle GoldenGate is reading. This option is not supported and causes Extract to abend.

---

---

In the following example, the command to set these attributes for a journal `JRN1` in library `LIB1` would be:

```
CHGJRN JRN(LIB1/JRN1) MNGRCV(*SYSTEM) DLTRCV(*NO) RCVSIZOPT(*MAXOPT2)
JRNSTATE(*ACTIVE) MINENTDTA(*NONE) FIXLENDTA(*USR *SYSSEQ)
```

---

---

**Note:** To check the attributes of a journal, use the command `WRKJRNA JRN(LIB1/JRN1) DETAIL(*CURATR)`.

---

---

When the journaling is set to the recommended parameter settings, you are assured that the entries in the journals contain all of the information necessary for Oracle GoldenGate processing to occur. These settings also ensure that the system does not delete the journal receivers automatically, but retains them in case Extract needs to process older data.

### 3.1.3 Deleting old journal receivers

Although the `DLTRCV` parameter is set to `NO` in the recommended configuration for Extract (see "[Setting journal parameters](#)" on page 3-1), you can delete old journal receivers manually once Extract is finished capturing from them.

#### To delete journal receivers

1. Run `GGSCI`.
2. In `GGSCI`, issue the following command to view the journal positions in which Extract has current processing points, along with their journal receivers.

```
INFO EXTRACT group name
```

3. Use the following `DB2 for i` command to delete any journal receivers that were generated prior to the ones that are shown in the `INFO EXTRACT` command.

```
DLTJRNRCV JRNRCV(library/journal_receiver)
```

Where:

*library* and *journal\_receiver* are the actual names of the library and journal receiver to be deleted. See the IBM i Information Center for more information about this command.

## 3.2 Specifying object names

Oracle GoldenGate commands and parameters support input in the form of SQL names, native names in the format of *library\_name/file\_name(member\_name)*, or a mix of the two. If a native file system name does not include the member name, all members are implicitly selected by the Oracle GoldenGate process

To support case sensitivity of double quoted object names, specify those names within double quotes in the Oracle GoldenGate parameter files. This is true of both SQL and native file system names.

For consistency of terminology in other administrative and reference Oracle GoldenGate documentation, the SQL terms "schema" and "table" are used to reference the containers for the DB2 for i data, as shown in [Table 3-1](#):

**Table 3-1 Native-SQL object name relationships**

Native	SQL	Notes
LIBRARY (maximum length 10)	SCHEMA (maximum length 128)	The operating system creates a corresponding native name for a SQL-created schema.
FILE (maximum length 10)	TABLE (maximum length 128)	The operating system creates a corresponding native name for a SQL-created table.
MEMBER	Not Applicable	Contains the actual data. Only the first member of a FILE object can be accessed through SQL. To access data in other members, either the native system name or a SQL alias to that name must be used.

## 3.3 Preparing tables for processing

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

### 3.3.1 Assigning row identifiers

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

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

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

1. Primary key
2. First unique key alphanumerically that does not contain a timestamp or non-materialized computed column.
3. If none of the preceding key types exist (even though there might be other types of keys defined on the table) Oracle GoldenGate constructs a pseudo key of all columns that the database allows to be used in a unique key, excluding those that are not supported by Oracle GoldenGate in a key or those that are excluded from the Oracle GoldenGate configuration.

---

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

---

### 3.3.1.2 Using KEYCOLS to specify a custom key

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

## 3.3.2 Preventing key changes

If you must add columns to the key that Extract is using as the row identifier for a table (primary key, unique key, `KEYCOLS` key, or all-column key) after Oracle GoldenGate has started processing journal data, follow these steps to make the change.

1. Stop Extract.
2. Issue the following command until it returns `EOF`, indicating that it has processed all of the existing journal data.

```
INFO EXTRACT group name
```

3. Make the change to the key.
4. Start Extract.

## 3.3.3 Disabling constraints on the target

Triggers and cascade constraints must be disabled on the target tables or configured to ignore changes made by Replicat. Constraints must be disabled because Oracle GoldenGate replicates DML that results from a trigger or a cascade constraint. If the same trigger or constraint gets activated on the target table, it becomes redundant because of the replicated version, and the database returns an error. Consider the following example, where the source tables are `emp_src` and `salary_src` and the target tables are `emp_targ` and `salary_targ`.

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

## 3.3.4 Enabling change capture

To capture changes to a table in a journal, you can run the `STRJRNPF` command on the OS/400 command line or run the `ADD TRANDATA` command from GGSCI. The `ADD TRANDATA` command calls `STRJRNPF` and is the recommended method to start journaling for tables, because it ensures that the required journal image attribute of Record Images (`IMAGES`): `*BOTH` is set on the `STRJRNPF` command.

### To run `ADD TRANDATA`

1. Run GGSCI on the source system.

**2. Issue the DBLOGIN command.**

```
DBLOGIN SOURCEDB database USERID user, PASSWORD password [encryption_options]
```

Where: SOURCEDB specifies the default DB 2 for i database. See the Oracle GoldenGate *Windows and UNIX Reference Guide* for password encryption options for DBLOGIN.

---

**Note:** Only BLOWFISH encryption is supported for DB2 for i systems.

---

**3. Issue the ADD TRANDATA command.**

```
ADD TRANDATA table_specification
```

Where: *table\_specification* is one of the following:

- *schema.table* [JOURNAL *library/journal*]
- *library/file* [JOURNAL *library/journal*] (See "[Specifying a default journal](#)")

**3.3.4.1 Specifying a default journal**

To specify a default journal for multiple tables or files in the ADD TRANDATA command, instead of specifying the JOURNAL keyword, use the following GGSCI command before issuing ADD TRANDATA.

```
DEFAULTJOURNAL library/journal
```

Any ADD TRANDATA command used without a journal assumes the journal from DEFAULTJOURNAL.

To display the current setting of DEFAULTJOURNAL, you can issue the command with no arguments.

**3.3.4.2 Removing a default journal specification**

To remove the use of a default journal, use the following GGSCI command:

```
DEFAULTJOURNAL CLEAR
```

**3.3.5 Maintaining materialized query tables**

To maintain parity between source and target materialized query tables (MQT), you replicate the base tables, but not the MQTs. The target database maintains the MQTs based on the changes that Replicat applies to the base tables.

The following are the rules for configuring these tables:

- Include the base tables in your TABLE and MAP statements.
- Do not include MQTs in the TABLE and MAP statements.
- Wildcards can be used in TABLE and MAP statements, even though they might resolve MQT names along with regular table names. Oracle GoldenGate automatically excludes MQTs from wildcarded table lists. However, any MQT that is explicitly listed in an Extract TABLE statement by name will cause Extract to abend.

### 3.3.6 Specifying the Oracle GoldenGate library

Before starting Oracle GoldenGate, specify the name of the Oracle GoldenGate for DB2 for i library when running the `ggos400install` script. This creates a link to the `OGGPRCJRN *SRVPGM` (service program) object that was restored to that library. If the link to the `oggprcjrnl` service program is deleted you could just re-run the `ggos400install` shell script and specify the same library, or use the command `"ln -s /qsys.lib/OGG_library.lib/oggprcjrnl.srvpgm oggprcjrnl.srvpgm"`. If this link is incorrect or missing, Extract will abend.

## 3.4 Adjusting the system clock

It is recommended that you set the system clock to UTC (Universal Time Coordinate) time and use the timezone offset in the IBM i system values to represent the correct local time. If this setup is done correctly, local daylight savings time adjustments can occur automatically with no disruption to replication.

## 3.5 Configuring the ODBC driver

The Replicat process cannot run locally on a DB2 for i series target system; it must run on a remote Windows or Linux system and connect to the target system over a database connection that is specified with ODBC. The following steps show how to install and configure ODBC to connect to the DB2 for i target system.

[Configuring ODBC on Linux](#)

[Configuring ODBC on Windows](#)

### 3.5.1 Configuring ODBC on Linux

To configure ODBC, you can use the graphical user interface to run the ODBC configuration utility that is supplied with your Linux distribution, or you can edit the `odbc.ini` file with the settings described in these steps. (These steps show the ODBC Administrator tool launched from the `ODBCConfig` graphical interface utility for Linux.)

1. Download and install the 32-bit or 64-bit iSeries Access ODBC driver on the remote Linux system according to the vendor documentation. The iSeries ODBC driver is supplied as a free component of iSeries Access.
2. Issue one of the following commands, depending on the driver that you want to use.

**32-bit driver:**

```
rpm -ivh iSeriesAccess-7.1.0-1.0.i386.rpm
```

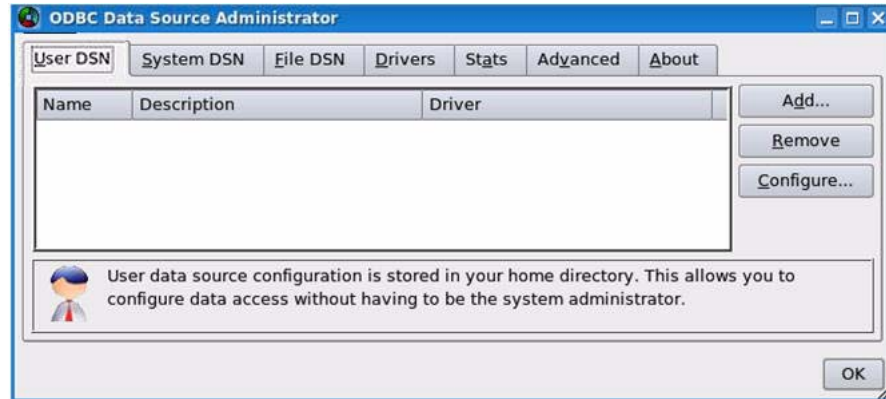
**64-bit driver:**

```
rpm -ivh iSeriesAccess-7.1.0-1.0.x86_64.rpm
```

3. You can create a user DSN (a connection that is available only to the user that created it) or a system DSN (a connection that is available to all users on the system). To create a user DSN, log on to the system as the user that you will be using for the Replicat process.
4. Run the ODBC configuration utility.

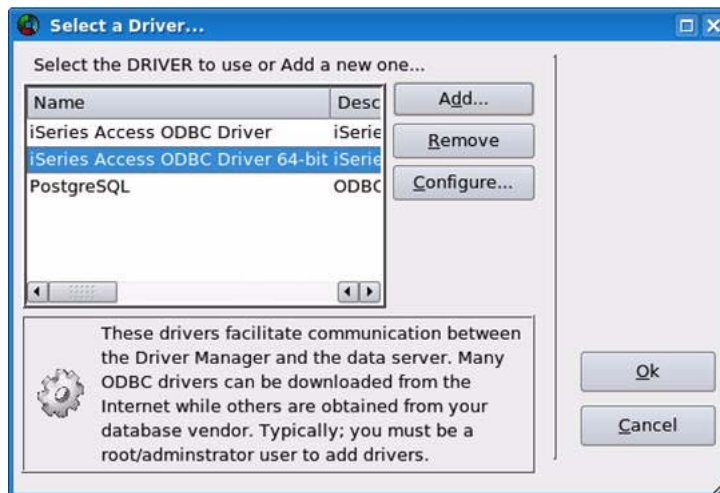
- On the initial page of the ODBC configuration tool, select the **User DSN** tab to create a user DSN or the **System DSN** tab to create a system DSN. (These steps create a user DSN; creating a system DSN is similar.)

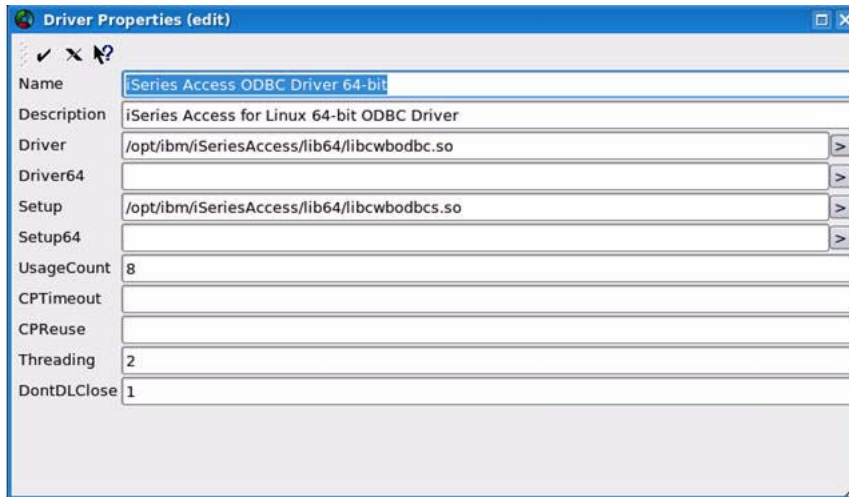
**Figure 3–1 Adding an ODBC DSN on Linux**



- On the tab you selected, click **Add**.
- Select the appropriate iSeries Access ODBC driver, click **OK**, and then go to step 8 of these instructions. If the correct driver is not shown in the **Select the DRIVER** list, click the **Add...** button and then complete the fields as shown in Figure 3–3.

**Figure 3–2 Selecting an ODBC driver on Linux**

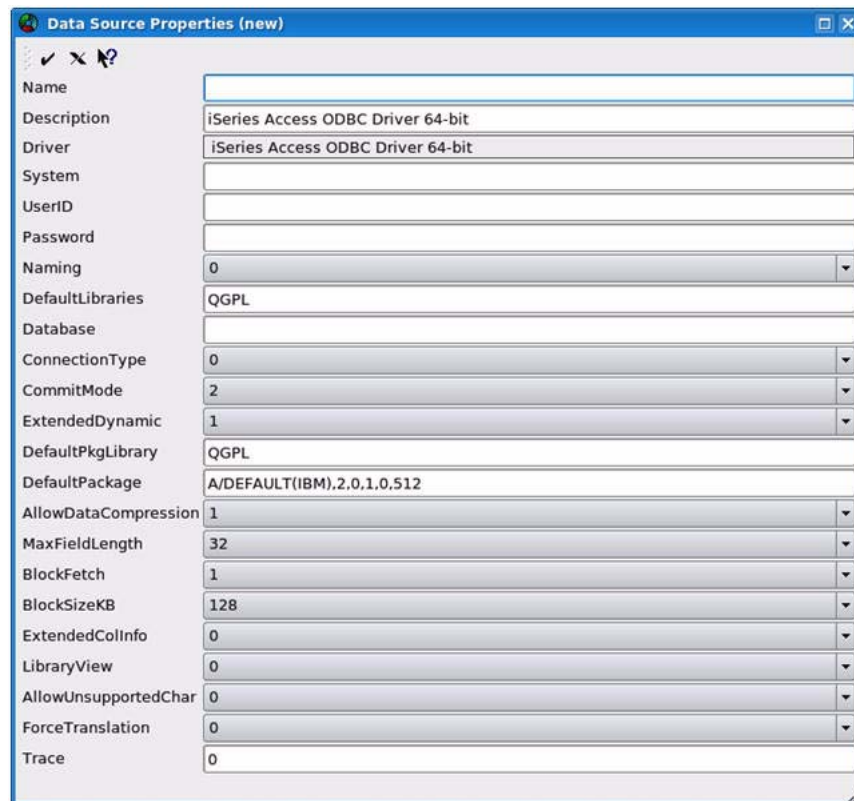


**Figure 3–3** Manually editing driver properties if the driver is not found

How to complete [Figure 3–3](#) fields:

- Set **Name** to the name of the driver.
  - Set **Driver** to the path where the driver is installed.
  - Set **Setup** to the `libcwbodbc.so` file that is in the driver installation directory.
  - Leave the other settings to their defaults.
  - Click the check mark above the **Name** field to save your settings.
8. In the **Name** field of the Data Source Properties dialog, supply a one-word name for the data source. In the **System** field, enter the fully qualified name of the target IBM i system, for example: `sysname.company.com`. Leave the **UserID** and **Password** fields blank, to allow Replicat to supply credentials when it connects to the database. Leave the remaining fields set to their defaults, and then click the check mark above the **Name** field to save your settings.

Figure 3–4 Setting Data Source properties



9. You are returned to the ODBC Data Source Administrator dialog. Click **OK** to exit the ODBC configuration utility.
10. To support GRAPHIC, VARGRAPHIC and DBLOB types, edit the `.odbc.ini` file and add the following line.

```
GRAPHIC = 1
```

---

**Note:** If you created a user Data Source Name, this file is located in the home directory of the user that created it. If you created a system DSN, this file is in `/etc/odbc.ini` or `/usr/local/etc/odbc.ini`.

---

11. From the Oracle GoldenGate directory on the target, run GGSCI and issue the `DBLOGIN` command to log into the target database.

```
DBLOGIN SOURCEDB database, USERID db_user [, PASSWORD pw [encryption options]]
```

Where:

- `SOURCEDB database` specifies the new Data Source Name.
- `USERID db_user`, `PASSWORD pw` are the Replicat database user profile and password.
- `encryption options` is optional password encryption. To use encryption, see the `USERID` parameter in the Oracle GoldenGate *Windows and UNIX Reference Guide*.

---



---

**Note:** Only BLOWFISH encryption is supported for DB2 for i systems.

---



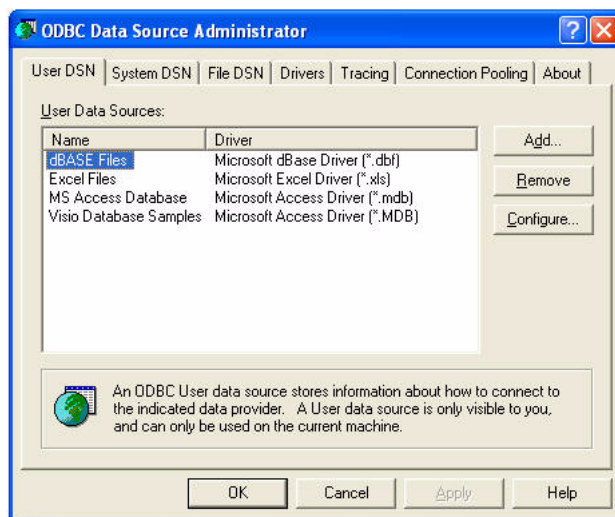
---

### 3.5.2 Configuring ODBC on Windows

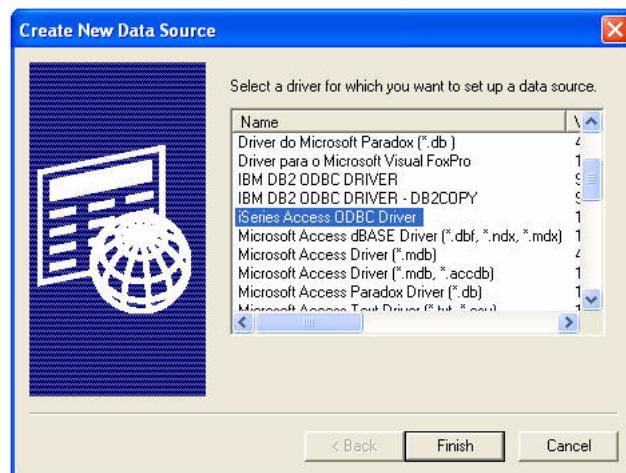
On Windows, the ODBC Administration tool is in the Administrative Tools folder as **Data Sources (ODBC)**.

1. Download and install the 32-bit or 64-bit iSeries Access ODBC driver from the IBM iSeries Access package on the remote Windows system according to the vendor documentation. The iSeries ODBC driver is supplied as a free component of iSeries Access.
2. You can create a user DSN (a connection that is available only to the user that created it) or a system DSN (a connection that is available to all users on the system). To create a user DSN, log on to the system as the user that you will be using for the Replicat process.
3. From the Windows Control Panel, select **Administrative Tools**, then **Data Sources (ODBC)**.
4. On the first page of the ODBC configuration tool, select the **User DSN** tab to create a user DSN or the **System DSN** tab to create a system DSN. (These steps create a user DSN; creating a system DSN is similar.)

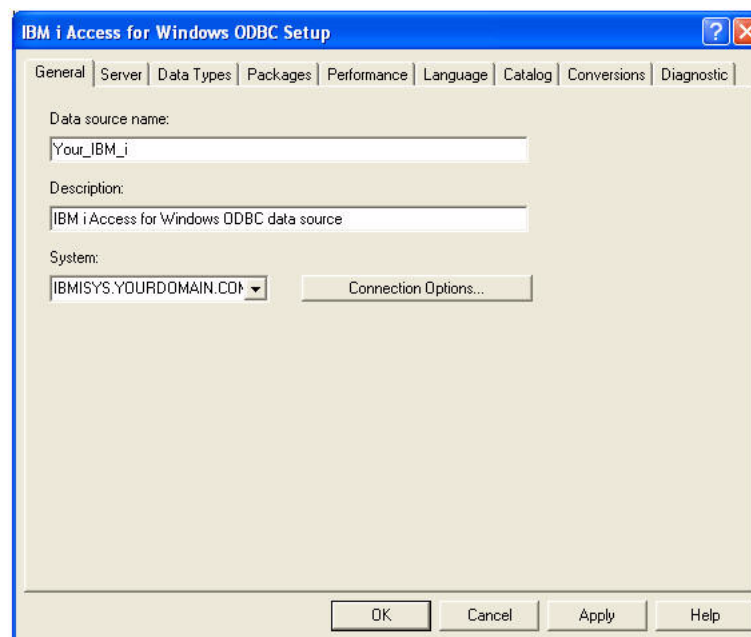
**Figure 3–5 Adding an ODBC DSN on Windows**



5. On the tab that you selected, click **Add**.
6. Select the appropriate iSeries Access ODBC Driver from the list of drivers, and then click **Finish**.

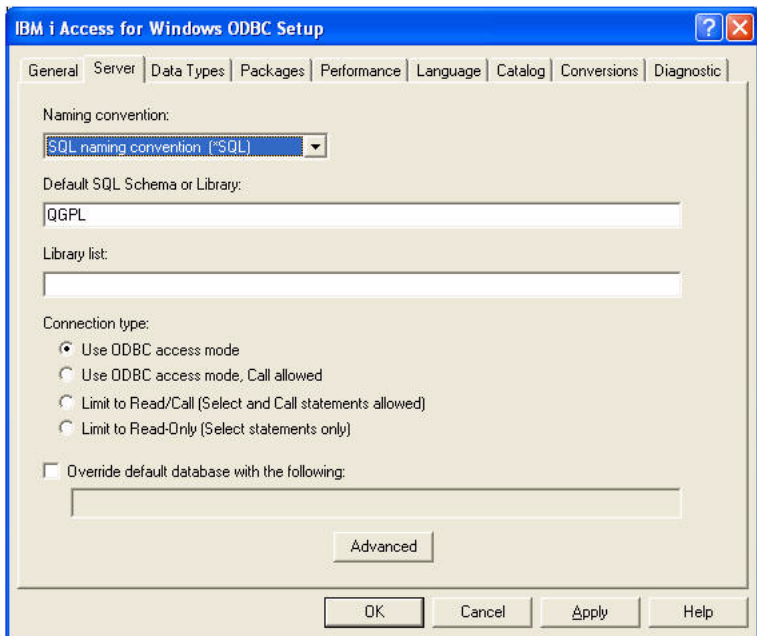
**Figure 3–6** Selecting an ODBC driver on Windows

7. On the General tab of the IBM i Access for Windows ODBC Setup dialog, provide a name (without any spaces) in the **Data Source Name** field, add an optional description in the **Description** field, and then select the system name from the **System** selection list.

**Figure 3–7** Setting general ODBC properties on Windows

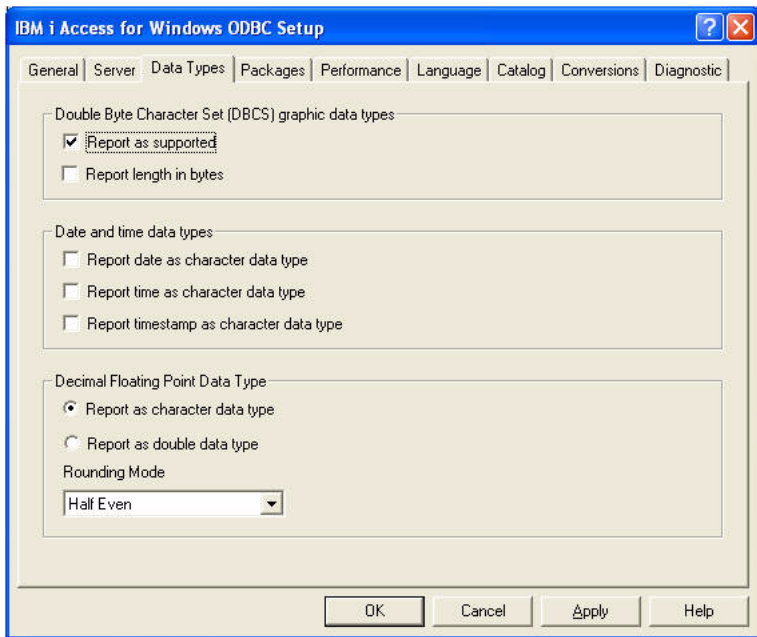
8. On the Server tab, set **Naming Convention** to SQL Naming Convention (\*SQL). Leave the other fields set to their defaults.

**Figure 3–8 Setting server ODBC properties on Windows**

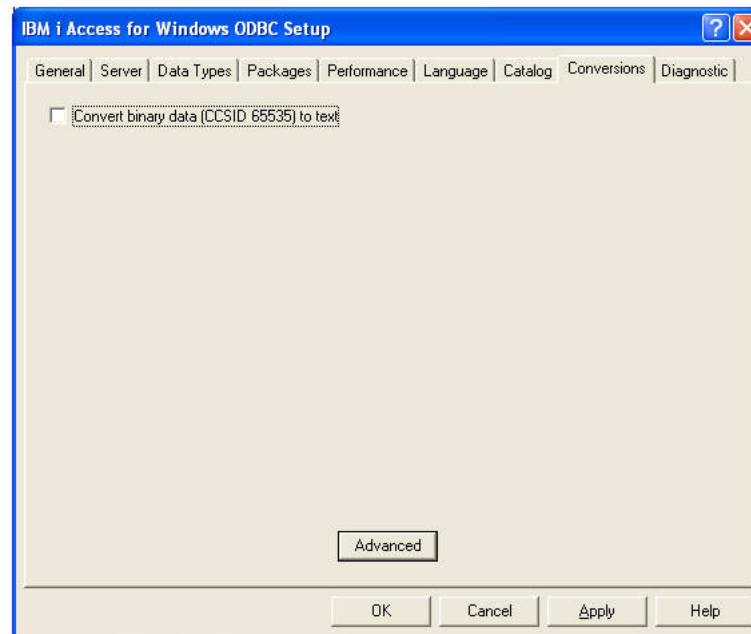


9. On the Data Types tab, select the **Report as Supported** check box under Double Byte Character Set (DBCS) graphic data types.

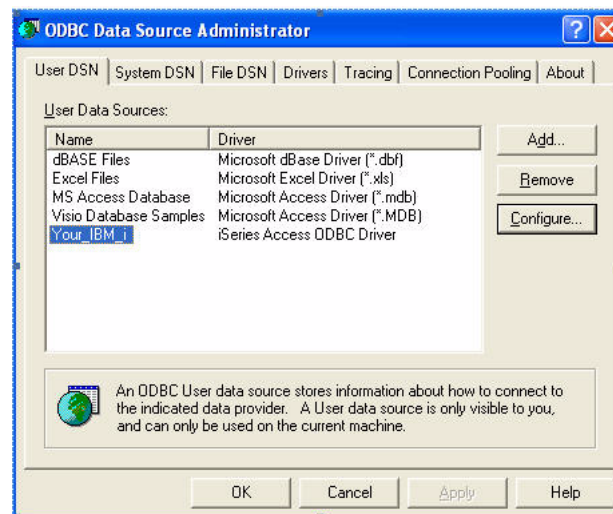
**Figure 3–9 Setting DBCS ODBC properties**



10. On the Conversions tab, clear the **Convert binary data (CCSID 65535) to text** check box.

**Figure 3–10 Setting Conversion ODBC properties**

11. Click **Apply**, then **OK**. You are returned to the ODBC Data Source Administrator dialog.
12. Confirm that the new Data Source Name appears under **User Data Sources**.

**Figure 3–11 Confirming new Data Source Name**

13. Click **OK** to exit the ODBC configuration utility.
14. From the Oracle GoldenGate directory on the target, run GGSCI and issue the DBLOGIN command to log into the target database.

```
DBLOGIN SOURCEDB database, USERID db_user [, PASSWORD pw [encryption options]]
```

Where:

- SOURCEDB *database* specifies the new data source name.

- USERID *db\_user*, PASSWORD *pw* are the Replicat database user profile and password.
- *encryption options* is optional password encryption. To use encryption, see the USERID parameter in the Oracle GoldenGate *Windows and UNIX Reference Guide*.

---

---

**Note:** Only BLOWFISH encryption is supported for DB2 for i systems.

---

---

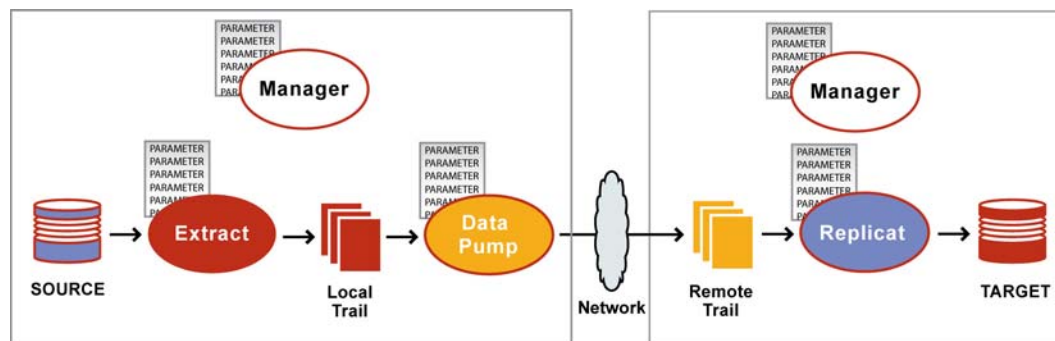
## Configuring Oracle GoldenGate on source and target databases

This chapter contains instructions for configuring Oracle GoldenGate to capture source DB2 for i data and apply it to a supported target database.

### 4.1 What to expect from this procedure

These instructions show you how to configure a basic set of Oracle GoldenGate parameter (configuration) files, one for each process that replicates transactional data changes from a DB2 for i source to a DB2 for i target or a different database type. Your business requirements probably will require a more complex topology, but this procedure forms a basis for the rest of your configuration steps.

**Figure 4–1 The basic configuration**



By performing these steps, you can:

- get the basic configuration files established.
- build upon them later by adding more parameters as you make decisions about features or requirements that apply to your environment.
- use copies of them to make additional parameter files faster than starting from scratch.

### 4.2 Creating and editing parameter files

You will be working with Oracle GoldenGate parameter files throughout the deployment process. To create and edit a parameter file, use the `EDIT PARAMS` command in GGSCI at any time after you start Manager.

```
EDIT PARAMS parameter_file_name
```

For more information about ways to work with Oracle GoldenGate parameter files, see the Oracle GoldenGate *Windows and UNIX Administrator's Guide*.

## 4.3 Overview of basic steps to configure Oracle GoldenGate

You will make the following decisions:

- [Choosing names for processes and files](#)

You will create the following basic set of objects.

### On both systems:

- [Creating the Oracle GoldenGate instance](#)

### On the source system:

- [Creating a data definitions file](#)
- [Configuring Extract for change capture](#)

### On the target system:

- [Configuring Replicat for change delivery](#)

This chapter also includes recommendations for:

- [Configuring additional process groups for best performance](#)
- [Next steps in the deployment](#)
- [When to start replicating transactional changes](#)
- [Testing your configuration](#)

## 4.4 Choosing names for processes and files

It is helpful to develop some naming conventions for the Oracle GoldenGate processes and files before you start configuration steps. Choosing meaningful names helps you differentiate among multiple processes and files in displays, error logs, and external monitoring programs. In addition, it accommodates the naming of additional processes and files later, as your environment changes or expands.

This section contains instructions for:

- [Choosing group names](#)
- [Choosing trail file names](#)

### 4.4.1 Choosing group names

You specify the names of Oracle GoldenGate processes in the parameter files and when you create them during the instantiation of the Oracle GoldenGate environment ([Chapter 8](#)). At minimum, you must choose names for the following processing components:

- **Primary Extract:** The primary Extract process captures data and DDL from the database source.
- **Data pump:** The data pump is a secondary Extract process that reads captured data from a local trail on the source system and sends it over the network to the target. The data pump adds storage flexibility and isolates the primary Extract

process from TCP/IP activity. For an illustration of how a data pump fits into the capture configuration, see [Figure 4-1](#).

- **Replicat:** The Replicat process reads a remote trail and applies the transactional changes to the target database.

Group names must be one word, up to eight alphanumeric characters that the operating system allows in a file name, including special characters except { \ / : \* ? " < > | }. Numbers may only be used at the beginning of a group name. For detailed information about process naming conventions, see the `ADD EXTRACT` and `ADD REPLICAT` commands for adding processes in the Oracle GoldenGate *Windows and UNIX Reference Guide*.

Once you evaluate your data, you may find it more beneficial to use more than one Replicat group and possibly more than one primary Extract group and data pump, depending on such decisions as capture method, performance tuning, or load balancing. Therefore, it is good practice to have a meaningful naming system.

For more information about using multiple process groups to tune the performance of Oracle GoldenGate, see the Oracle GoldenGate *Windows and UNIX Troubleshooting and Tuning Guide*.

## 4.4.2 Choosing trail file names

Captured data must be processed into a series of files called a trail, where it is stored for processing by the next Oracle GoldenGate process downstream. The basic configuration is:

- a local trail on the source system
- a remote trail on the target system.

By default, trail files are stored in the `dirdat` subdirectory of the Oracle GoldenGate installation directory. The name of a trail file includes its directory path plus the actual trail name: a two-character designation such as the "tr" in `./dirdat/tr`. In the event that you must use more than one set of local and remote trails, it is a good practice to devise a meaningful set of two-character naming conventions as you create the trails during the configuration process.

---

**Note:** The names of the actual files in the trail consist of the trail name appended with a six-digit sequence number. During processing, the sequence number automatically increments by one whenever a new file is created, such as `./dirdat/tr000001`, `./dirdat/tr000002`, and so forth.

---

For more information about trail files, how they are stored, and how they are managed, see the Oracle GoldenGate *Windows and UNIX Administrator's Guide*.

## 4.5 Creating the Oracle GoldenGate instance

Each Oracle GoldenGate installation is rooted in the Manager process. This is the controller process that instantiates the Oracle GoldenGate processes, allocates port numbers, and performs file maintenance. Together, the Manager process and its child processes, and their related programs and files comprise an Oracle GoldenGate instance.

To run Oracle GoldenGate, a Manager process must be running on all systems that will be part of the Oracle GoldenGate environment. To run Manager, you first create a parameter file for it.

### To create the Manager parameter file

1. From the Oracle GoldenGate directory, run the ggscli program to open the Oracle GoldenGate Software Command Interface (GGSCI).
2. In GGSCI, edit the Manager parameter file.

```
EDIT PARAMS MGR
```

3. Add the Manager parameters, each on one line. If a parameter statement must span multiple lines, use an ampersand (&) before each line break.
  - The only required Manager parameter is `PORT`, but `DYNAMICPORTLIST` is strongly recommended.
  - In a cluster environment, configure Manager with the `AUTOSTART` and `AUTORESTART` parameters, so that the Oracle GoldenGate processes start or restart automatically when Manager is started or fails over.
  - Use `PURGEOLDEXTRACTS` to manage the accumulation of trail files.
  - For more information about Manager parameters, see the Oracle GoldenGate *Windows and UNIX Reference Guide*.

For more details on configuring Manager and its network connections, see the Oracle GoldenGate *Windows and UNIX Administrator's Guide*.

4. Save, then close the file.

#### Example 4-1

The following is a sample Manager parameter file (showing Linux file structure; use appropriate path notation for your operating system).

```
PORT 7809
DYNAMICPORTLIST 7810-7820, 7830
AUTOSTART ER t*
AUTORESTART ER t*, RETRIES 4, WAITMINUTES 4
STARTUPVALIDATIONDELAY 5
PURGEOLDEXTRACTS /ogg/dirdat/tt*, USECHECKPOINTS, MINKEEPHOURS 2
```

## 4.6 Creating a data definitions file

When replicating data from one table to another, an important consideration is whether the column structures (metadata) of the source and target tables are identical. Oracle GoldenGate looks up metadata for the following purposes:

- On the source, to supply complete information about captured operations to the Replicat process.
- On the target, to determine the structures of the target tables, so that the replicated data is correctly mapped and converted (if needed) by Replicat.

When source and target table definitions are dissimilar, Oracle GoldenGate must perform a conversion from one format to the other. To perform conversions, both sets of definitions must be known to Oracle GoldenGate. Oracle GoldenGate can query the local database to get one set of definitions, but it must rely on a *data-definitions file* to get definitions from the remote database. The data-definitions file contains information about the metadata of the data that is being replicated.

To create a definitions file, you configure and run the DEFGEN utility and then transfer the definitions file to the target system. For instructions, see "Associating replicated data with metadata" in the Oracle GoldenGate *Windows and UNIX Administrator's Guide*. This file must be in place on the target system before you start the Oracle GoldenGate processes for the first time.

## 4.7 Configuring Extract for change capture

Perform these steps on the source system to configure the primary and data pump Extract processes that support change capture and transport across the network.

### 4.7.1 Encrypting the Extract passwords

These steps encrypt the Extract user profile's password for both the primary and data pump Extracts. It is strongly recommended that you encrypt all passwords. Oracle GoldenGate supports Blowfish encryption on the DB2 for i platform.

1. In GGSCI on the source system, issue the following command.

```
ENCRYPT PASSWORD password, BLOWFISH ENCRYPTKEY keyname
```

Where: *password* is the clear-text password, and BLOWFISH ENCRYPTKEY *keyname* specifies Blowfish encryption with the name of a logical key that can be used to look up an actual key stored in a local ENCKEYS file. To use encryption, you must generate a key and create the ENCKEYS file at some point before you start processes. For more information, see the Oracle GoldenGate *Windows and UNIX Administrator's Guide*.

2. Keep a record of the encrypted key by copying and pasting it into a text file or leaving the GGSCI session open, so that it is available to copy into the parameter files.

### 4.7.2 Configuring the primary Extract

These steps configure the Extract that captures the transaction data.

1. In GGSCI on the source system, create the Extract parameter file.

```
EDIT PARAMS name
```

Where: *name* is the name of the primary Extract. group. See "[Specifying object names](#)" on page 3-2 for naming conventions.

2. Enter the Extract parameters in the order shown, starting a new line for each parameter statement. Your input variables will be different. See [Table 4-1](#) for descriptions.

#### Basic parameters for the primary Extract group:

```
EXTRACT finance
SOURCEDE FINANCE USERID ogg, PASSWORD AACAAAAAAAAAAAA, BLOWFISH ENCRYPTKEY mykey
ENCRYPTTRAIL BLOWFISH KEYNAME mykey2
EXTTRAIL /ggs/dirdat/lt
TABLE hr.*;
```

**Table 4–1 Basic parameters for the primary Extract**

Parameter	Description
EXTRACT <i>group name</i>	<i>group name</i> is the name of the Extract group. See "Specifying object names" on page 3-2 for naming conventions.
SOURCEDB <i>database id</i> , USERID <i>user id</i> , PASSWORD <i>password</i> , BLOWFISH ENCRYPTKEY <i>keyname</i>	Specifies database connection information. <ul style="list-style-type: none"> <li>SOURCEDB specifies the <i>default</i> DB 2 for i database that is identified by the system name (in upper case). Only this database is supported. Named databases on other independent auxiliary storage pools (IASPs) are not supported.</li> <li>USERID specifies the Extract database user profile.</li> <li>PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command. Enter or paste the encrypted password after the PASSWORD keyword.</li> <li>BLOWFISH ENCRYPTKEY <i>keyname</i> specifies the name of the lookup key in the local ENCKEYS file.</li> </ul>
ENCRYPTTRAIL BLOWFISH KEYNAME <i>keyname</i>	Encrypts the local trail with Blowfish encryption. For more information, see the Oracle GoldenGate security documentation in the <i>Windows and UNIX Administrator's Guide</i> .
EXTRAIL <i>pathname</i>	Specifies the path name of the local trail to which the primary Extract writes captured data.
TABLE <i>owner.table</i> ;	Specifies a table or tables for which to extract data changes. See "Specifying object names" on page 3-2 for supported ways to specify object names. Only table names be wildcarded. To extract data from tables in multiple schemas or libraries, use a separate TABLE statement for each one. Terminate the TABLE statement with a semi-colon. To exclude tables from a wildcard specification, use the TABLEEXCLUDE parameter after the TABLE statement. For more information about parameters and options that control data filtering, mapping, and manipulation, see the Oracle GoldenGate <i>Windows and UNIX Reference Guide</i> .

- Enter any optional Extract parameters that are recommended elsewhere in this manual and any others shown in the Oracle GoldenGate *Windows and UNIX Reference Guide*.
- Save and close the file.

### 4.7.3 Configuring the data pump

These steps configure the data pump that reads the local trail and sends the data across the network to a remote trail.

- In GGSCI on the source system, create the data-pump parameter file.

```
EDIT PARAMS name
```

Where: *name* is the name of the data pump Extract group. See "Specifying object names" on page 3-2 for naming conventions.

- Enter the data-pump parameters in the order shown, starting a new line for each parameter statement. Your input variables will be different. See Table 4–2 for descriptions.

#### Basic parameters for the data-pump Extract group:

```
EXTRACT extpump
```

```
SOURCEDB FINANCE USERID ogg, PASSWORD AACAAAAAAAAAAAA, BLOWFISH ENCRYPTKEY mykey
RMTHOST fin1, MGRPORT 7809
RMTTRAIL /ggs/dirdat/rt
TABLE hr.*;
```

**Table 4–2 Basic parameters for the data-pump Extract**

Parameter	Description
EXTRACT <i>group name</i>	<i>group name</i> is the name of the data pump. See "Specifying object names" on page 3-2 for naming conventions.
SOURCEDB <i>database</i> USERID <i>user id</i> , PASSWORD <i>password</i> , BLOWFISH ENCRYPTKEY <i>keyname</i>	Specifies database connection information. <ul style="list-style-type: none"> <li>■ SOURCEDB specifies the <i>default</i> DB 2 for i database.</li> <li>■ USERID specifies the Extract database user profile.</li> <li>■ PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command. Enter or paste the encrypted password after the PASSWORD keyword.</li> <li>■ BLOWFISH ENCRYPTKEY <i>keyname</i> specifies the name of the lookup key in the local ENCKEYS file.</li> </ul>
DECRYPTTRAIL BLOWFISH KEYNAME <i>keyname</i>	Decrypts the input trail.
RMTHOST <i>hostname</i> , MGRPORT <i>portnumber</i>	<ul style="list-style-type: none"> <li>■ RMTHOST specifies the name or IP address of the target system.</li> <li>■ MGRPORT specifies the port number where Manager is running on the target.</li> </ul> For additional options and encryption details, see the Oracle GoldenGate <i>Windows and UNIX Reference Guide</i> .
ENCRYPTTRAIL BLOWFISH KEYNAME <i>keyname</i>	Encrypts the remote trail with Blowfish encryption. For more information, see the Oracle GoldenGate security documentation in the <i>Windows and UNIX Administrator's Guide</i> .
RMTTRAIL <i>pathname</i>	Specifies the path name of the remote trail.
TABLE <i>owner.table</i> ;	Specifies a table or tables. In most cases, this listing will be the same as that in the primary Extract parameter file. <ul style="list-style-type: none"> <li>■ <i>owner</i> is the schema name.</li> <li>■ <i>table</i> is the name of the table or a group of tables defined with wildcards.</li> </ul> Schema names cannot be wildcarded. To extract data from tables in multiple schemas, use a separate TABLE statement for each schema. For example: <pre>TABLE fin.*; TABLE hr.*;</pre> Terminate the TABLE statement with a semi-colon. To exclude tables from a wildcard specification, use the TABLEEXCLUDE <i>owner.table</i> parameter after the TABLE statement. For more information and for additional options that control data filtering, mapping, and manipulation, see TABLE in the Oracle GoldenGate <i>Windows and UNIX Reference Guide</i> .

3. Enter any optional Extract parameters that are recommended elsewhere in this manual and any others shown in the Oracle GoldenGate *Windows and UNIX Reference Guide*.
4. Save and close the file.

## 4.8 Configuring Replicat for change delivery

Perform these steps on the intermediary Windows or Linux system on which Oracle GoldenGate is installed to communicate remotely with the target over ODBC. These steps configure the objects that support change delivery to the target database.

---

---

**Note:** To configure Replicat for change delivery to a non-DB2 for i database, such as an Oracle database, follow the directions in the Oracle GoldenGate *Installation and Setup* guide for that database. There may be additional parameters and requirements for delivery to that database type.

---

---

### 4.8.1 Creating a checkpoint table

Replicat maintains its checkpoints in a checkpoint table in the target database. Each checkpoint is written to the checkpoint table within the Replicat transaction. Because a checkpoint either succeeds or fails with the transaction, Replicat ensures that a transaction is only applied once, even if there is a failure of the process or the database.

---

---

**Note:** This procedure installs a default checkpoint table, which is sufficient in most cases. For details, see the `CHECKPOINTTABLE` option of `ADD REPLICAT` in the Oracle GoldenGate *Windows and UNIX Reference Guide*.

---

---

#### 4.8.1.1 Adding the checkpoint table to the target database

Create the checkpoint table in the target database prior to creating Replicat groups.

1. From the Oracle GoldenGate directory on the target, run GGSCI and issue the `DBLOGIN` command to log into the target database.

```
DBLOGIN SOURCEDB database, USERID db_user [, PASSWORD pw [encryption options]]
```

Where:

- `SOURCEDB database` specifies the default DB 2 for i database that is identified by the system name (in upper case).
- `USERID db_user`, `PASSWORD pw`, and `encryption options` supply database credentials of a user with `CREATE TABLE` permissions and optional password encryption. See the Oracle GoldenGate *Windows and UNIX Reference Guide* for more information about command options.

---

---

**Note:** Only `BLOWFISH` encryption is supported for DB2 for i systems.

---

---

2. In GGSCI, create the checkpoint table in a schema of your choice (ideally dedicated to Oracle GoldenGate).

```
ADD CHECKPOINTTABLE owner.table
```

#### 4.8.1.2 Specifying the checkpoint table in the Oracle GoldenGate configuration

Specify the name of the checkpoint table in the `GLOBALS` parameter file.

1. Create a `GLOBALS` file (or edit the existing one).

```
EDIT PARAMS ./GLOBALS
```

---

**Note:** GLOBALS must be in upper case, without a file extension. The file must remain in the root Oracle GoldenGate directory.

---

2. In the GLOBALS file, enter the CHECKPOINTTABLE parameter.

```
CHECKPOINTTABLE owner.table
```

Where: *owner.table* is the owner and a name that is supported by the database.

3. Save and close the GLOBALS file.

## 4.8.2 Encrypting the Replicat password

These steps encrypt the Replicat user profile's password. It is strongly recommended that you encrypt all passwords. Oracle Golden supports Blowfish encryption on the DB2 for i platform. On other target platforms, there is the additional option of Advanced Encryption Security (AES). For more information about Oracle GoldenGate security options, see the Oracle GoldenGate *Windows and UNIX Administrator's Guide*.

1. In GGSCI on the target system, issue the following command.

```
ENCRYPT PASSWORD password algorithm ENCRYPTKEY keyname
```

Where:

- *password* is the clear-text password.
  - *algorithm* is BLOWFISH for a DB2 for i target, or one of the following for a non-DB2 for i target: AES128, AES192, or AES256.
  - ENCRYPTKEY *keyname* is the name of the lookup key in the ENCKEYS file.
2. Keep a record of the encrypted key by copying and pasting it into a text file or leaving the GGSCI session open, so that it is available to copy into the parameter file.

## 4.8.3 Configuring Replicat

These steps configure the Replicat process in a basic way without any special mapping or conversion of the data. For more advanced mapping options, see the Oracle GoldenGate *Windows and UNIX Administrator's Guide*.

1. In GGSCI on the target system, create the Replicat parameter file.

```
EDIT PARAMS name
```

Where: *name* is the name of the Replicat group. See ["Specifying object names"](#) on page 3-2 for naming conventions.

2. Enter the Replicat parameters in the order shown, starting a new line for each parameter statement. See [Table 4-3](#) for descriptions.

```
REPLICAT financier
TARGETDB FINANCIAL USERID ogg, PASSWORD AACAAAAAAAAAAAA, BLOWFISH ENCRYPTKEY
mykey
ASSUMETARGETDEFS
-- Instead of ASSUMETARGETDEFS, use SOURCEDEFS if replicating from
-- DB2 for i to a different database type, or from a DB2 for i source
-- that is not identical in definitions to a target DB2 for i database.
```

```
-- SOURCEDEFS /users/ogg/dirdef/defsfile
DISCARDFILE /users/ogg/disc
MAP hr.*, TARGET hr2.*;
```

**Table 4–3 Basic parameters for Replicat**

Parameter	Description
REPLICAT <i>group name</i>	<i>group name</i> is the name of the Replicat group. See "Specifying object names" on page 3-2 for naming conventions.
TARGETDB <i>database</i> , USERID <i>user id</i> , PASSWORD <i>pw</i> , <i>algorithm</i> ENCRYPTKEY <i>keyname</i>	Specifies database connection information. <ul style="list-style-type: none"> <li>TARGETDB specifies the correct Data Source Name that is defined for the DB2 for i target database through the ODBC interface on the Windows or Linux system.</li> <li>USERID specifies the Replicat database user profile.</li> <li>PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command. Enter or paste the encrypted password after the PASSWORD keyword.</li> <li><i>algorithm</i> ENCRYPTKEY <i>keyname</i> specifies the encryption method and keyname that was specified in ENCRYPT PASSWORD.</li> </ul>
DECRYPTTRAIL BLOWFISH KEYNAME <i>keyname</i>	Decrypts the input trail.
SOURCEDEFS <i>pathname</i>   ASSUMETARGETDEFS	Specifies how to interpret data definitions. Use SOURCEDEFS if the source and target tables have different definitions, such as when replicating data between dissimilar IBM for i databases or from an IBM for i database to an Oracle database. For <i>pathname</i> , specify the source data-definitions file that you created with the DEFGEN utility in "Creating a data definitions file" on page 4-4. Use ASSUMETARGETDEFS if the source and target tables are all DB2 for i and have the same definitions.
DISCARDFILE <i>full_</i> <i>pathname</i>	Specifies the full path name of a file to which Replicat writes rejected record data, for example records that generated database errors. A discard file is optional but recommended. See the Oracle GoldenGate <i>Windows and UNIX Reference Guide</i> for additional options.
MAP <i>owner.table</i> , TARGET <i>owner.table</i> ;	Specifies a relationship between a source and target table or tables. <ul style="list-style-type: none"> <li><i>owner</i> is the schema or library name.</li> <li><i>table</i> is the name of a table or a wildcard definition for multiple tables.</li> </ul> <p>Schema or library names cannot be wildcarded. To map tables in multiple schemas or libraries, use a separate MAP statement for each one. For example:</p> <pre>MAP fin.*, TARGET fin.*; MAP hr.*, TARGET hr.*;</pre> <p>Terminate the MAP statement with a semi-colon.</p> <p>To exclude tables from a wildcard specification, use the MAPEXCLUDE parameter.</p> <p>For more information and for additional options that control data filtering, mapping, and manipulation, see MAP in the Oracle GoldenGate <i>Windows and UNIX Reference Guide</i>.</p>

3. Enter any optional Extract parameters that are recommended elsewhere in this manual and any others shown in the Oracle GoldenGate *Windows and UNIX Reference Guide*.
4. Save and close the file.

## 4.9 Configuring additional process groups for best performance

Develop business rules that specify the acceptable amount of lag between the time when transactions are generated by your source applications and when those changes

are applied to the target database. These rules determine the number of parallel Extract and Replicat processes that are required to enable Oracle GoldenGate to perform at its best.

Gather the size and activity rates of all of the tables that you intend to replicate with Oracle GoldenGate, and then:

- Assign one Extract group to the tables with low activity rates.
- Assign a dedicated Extract group to tables with high activity rates.

Configure these Extract groups to work with dedicated data pumps and Replicat groups. Once you have a basic set of parameter files for one Extract and Replicat process, you can:

- copy them to the names of the new process groups.
- edit the copies to produce the parameter files for your additional process groups.

For more information about configuring Oracle GoldenGate for best performance, see the Oracle GoldenGate *Windows and UNIX Troubleshooting and Tuning Guide*.

## 4.10 Oracle GoldenGate parameters not supported for DB2 for i

This section lists the Oracle GoldenGate configuration parameters that are not supported for the DB2 for i platform.

ASCIITOEBCDIC and EBCDICTOASCII  
 BINARYCHARS  
 CHARSET  
 CHARSETCONVERSION  
 LOBMEMORY  
 TRAILCHARSETEBCDIC  
 Any encryption options that use AES encryption

## 4.11 Next steps in the deployment

Because of its flexibility, Oracle GoldenGate offers numerous features and options that must be considered before you start any processes. To further configure Oracle GoldenGate to suit your business needs, see the following:

- For additional configuration guidelines to achieve specific replication topologies, see the Oracle GoldenGate *Windows and UNIX Administrator's Guide*. This guide also contains information about:
  - Oracle GoldenGate architecture
  - Oracle GoldenGate commands
  - Oracle GoldenGate initial load methods
  - Configuring security
  - Using customization features
  - Mapping columns that contain dissimilar data
  - Data filtering and manipulation
- For syntax options and descriptions of Oracle GoldenGate GGSCI commands and Oracle GoldenGate parameters shown in this guide, see the Oracle GoldenGate *Windows and UNIX Reference Guide*.

## 4.12 When to start replicating transactional changes

You must start replication when the source and target data is in a synchronized state, where the corresponding rows in the source and target tables contain identical data values. Unless you are starting with brand new source and target databases with no current user activity, you will need to activate change synchronization to capture ongoing transactional changes while an initial load is being applied to the target. The initial load captures a point-in-time snapshot of the source data and applies it to the target, while Oracle GoldenGate maintains any changes that are made after that point.

After you satisfy the requirements for configuring the database and the Oracle GoldenGate processes (including DDL support if required), see "[Instantiating and starting Oracle GoldenGate replication](#)" on page 8-1 to perform an initial load, start replicating changes, and handle post-load collisions.

## 4.13 Testing your configuration

It is important to test your configuration in a test environment before deploying it live on your production machines. This is especially important in an active-active or high availability configuration, where trusted source data may be touched by the replication processes. Testing enables you to find and resolve any configuration mistakes or data issues without the need to interrupt user activity for re-loads on the target or other troubleshooting activities.

---

---

# Instantiating and starting Oracle GoldenGate replication

This chapter contains instructions for configuring an initial load of target data, adding the required processes to instantiate replication, and perform the instantiation. The expected outcome of these steps is that source-target data is made consistent (known as the initial synchronization), and that Oracle GoldenGate captures and delivers ongoing transactional changes so that consistency is maintained going forward.

## 5.1 About the instantiation process

During the initialization of the Oracle GoldenGate environment, you will be doing an initial data synchronization and starting the Oracle GoldenGate processes for the first time. In conjunction with those procedures, you will be creating the process groups for which you created parameter files in [Chapter 4](#).

To create an Extract process group, an initial start position for data capture must be established. This initial position will be based on a transaction boundary that is based on either of the following:

- a timestamp
- the end of the journal(s)
- A specific system sequence number
- A specific sequence number in the journal(s)

When Extract starts for the first time to begin capturing data, it captures all of the transaction data that it encounters after the specified start point, but none of the data that occurred before that point. To ensure that Extract does not start in the middle of ongoing transactions that would be discarded, set the tables that are to be captured to an inactive state. You can either put the system into a restricted state by using the `ALCOBJ` command to lock the objects or libraries, or you can force all of the current transactions on those tables to stop at a certain point.

After initialization is complete, remember to unlock any objects that you locked. To do so, log off of the session that locked the objects or use the `DLCOBJ` command from the `OS/400` command line.

## 5.2 Overview of basic Oracle GoldenGate instantiation steps

These instructions show you how to instantiate the basic replication environment that you configured in Chapter 4. These steps are:

- [Satisfying prerequisites for instantiation](#)

- [Making the instantiation procedure more efficient](#)
- [Configuring the initial load](#)
- [Adding change-capture and change-delivery processes](#)
- [Performing the target instantiation](#)
- [Monitoring processing after the instantiation](#)
- [Backing up the Oracle GoldenGate environment](#)
- [Positioning Extract after startup](#)

## 5.3 Satisfying prerequisites for instantiation

These steps must be taken before starting any Oracle GoldenGate processes or native database load processes.

### 5.3.1 Configure change capture and delivery

By the time you are ready to instantiate the replication environment, all of your Extract and Replicat process groups must be configured with completed parameter files as directed in ["Configuring Oracle GoldenGate on Oracle source and target databases"](#) on page 4-1.

In addition, all of the other setup requirements in this manual must be satisfied.

### 5.3.2 Add collision handling

If the source database will remain active during the initial load, collision-handling logic must be added to the Replicat parameter file. This logic handles conflicts that occur because static data is being loaded to the target tables while Oracle GoldenGate replicates transactional changes to those tables.

To handle collisions, add the `HANDLECOLLISIONS` parameter to the Replicat parameter file to resolve:

- `INSERT` operations for which the row already exists.
- `UPDATE` and `DELETE` operations for which the row does not exist.

For more information about this parameter, see the Oracle GoldenGate *Windows and UNIX Reference Guide*.

### 5.3.3 Prepare the target tables

The following are suggestions that can make the load go faster and help you to avoid errors.

- **Data:** Make certain that the target tables are empty. Otherwise, there may be duplicate-row errors or conflicts between existing rows and rows that are being loaded.
- **Constraints:** If you have not done so already, disable foreign-key constraints and check constraints. Foreign-key constraints can cause errors, and check constraints can slow down the loading process.
- **Indexes:** Remove indexes from the target tables. Indexes are not necessary for the inserts performed by the initial load process and will slow it down significantly. You can add back the indexes after the load is finished.

- **Keys:** To use the `HANDLECOLLISIONS` function to reconcile incremental data changes with the load, each target table must have a primary or unique key. If you cannot create a key through your application, use the `KEYCOLS` option of the `TABLE` and `MAP` parameters to specify columns as a substitute key for Oracle GoldenGate's purposes. If you cannot create keys, the affected source table must be quiesced for the load.

## 5.4 Making the instantiation procedure more efficient

The following are some suggestions for making the instantiation process move more efficiently.

### 5.4.1 Share parameters between process groups

Some of the parameters that you use in a change-synchronization parameter file also are required in an initial-load Extract and initial-load Replicat parameter file. To take advantage of the commonalities, you can use any of the following methods:

- Copy common parameters from one parameter file to another.
- Store the common parameters in a central file and use the `OBEY` parameter in each parameter file to retrieve them.
- Create an Oracle GoldenGate macro for the common parameters and then call the macro from each parameter file with the `MACRO` parameter.

### 5.4.2 Use parallel processes

You can configure parallel initial-load processes to perform the initial load more quickly. It is important to keep tables with foreign-key relationships within the same set of processes. You can isolate large tables from smaller ones by using different sets of processes, or simply apportion the load across any number of process sets. To configure parallel processes correctly, see the *Oracle GoldenGate Windows and UNIX Troubleshooting and Tuning Guide*.

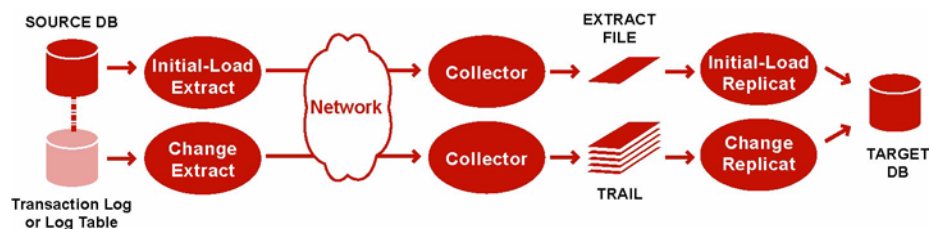
## 5.5 Configuring the initial load

Oracle GoldenGate supports the following load methods specifically for Oracle:

- [Configuring an initial load from file to Replicat](#)
- [Configuring an initial load with a database utility](#)

Select a method and follow its configuration steps to create the load processes and parameter files.

### 5.5.1 Configuring an initial load from file to Replicat



To use Replicat to establish the target data, you use an initial-load Extract to extract source records from the source tables and write them to an extract file in canonical format. From the file, an initial-load Replicat loads the data using the database interface. During the load, the change-synchronization groups extract and replicate incremental changes, which are then reconciled with the results of the load.

During the load, the records are applied to the target database one record at a time, so this method may be considerably slower than using a native IBM i load utility. This method permits data transformation to be done on either the source or target system.

**To configure a load from file to Replicat**

1. On the source and target systems, run GGSCI and start Manager.

```
START MANAGER
```

2. On the source system, issue the following command to create an initial-load Extract parameter file. This Extract should have a different name from the Extract groups that capture the transactional data.

```
EDIT PARAMS initial-load Extract name
```

3. Enter the parameters listed in [Table 5-1](#) in the order shown, starting a new line for each parameter statement.

**Table 5-1 Initial-load Extract parameters for loading data from file to Replicat**

Parameter	Description
SOURCEISTABLE	Designates Extract as an initial load process extracting records directly from the source tables.
SOURCEDB <i>database</i> USERID <i>user id</i> , PASSWORD <i>password</i> , BLOWFISH ENCRYPTKEY <i>keyname</i>	<p>Specifies database connection information.</p> <ul style="list-style-type: none"> <li>■ SOURCEDB specifies the name of the source database.</li> <li>■ USERID specifies the Extract database user profile.</li> <li>■ PASSWORD specifies the user’s password that was encrypted with the ENCRYPT PASSWORD command (see <a href="#">"Encrypting the Extract passwords"</a> on page 4-5). Enter or paste the encrypted password after the PASSWORD keyword.</li> <li>■ BLOWFISH ENCRYPTKEY <i>keyname</i> specifies the name of the lookup key in the local ENCKEYS file.</li> </ul>
RMTHOST <i>hostname</i> , MGRPORT <i>portnumber</i> , [ <i>encryption options</i> ]	<ul style="list-style-type: none"> <li>■ RMTHOST specifies the name or IP address of the target system.</li> <li>■ MGRPORT specifies the port number where Manager is running on the target.</li> <li>■ <i>encryption options</i> specifies optional encryption of data across TCP/IP.</li> </ul> <p>For additional options and encryption details, see the <i>Oracle GoldenGate Windows and UNIX Reference Guide</i>.</p>

**Table 5–1 (Cont.) Initial-load Extract parameters for loading data from file to Replicat**

Parameter	Description
ENCRYPTTRAIL BLOWFISH KEYNAME <i>keyname</i>	Encrypts the remote file with Blowfish encryption. For more information, see the Oracle GoldenGate security documentation in the <i>Windows and UNIX Administrator's Guide</i> .
RMTFILE <i>path name</i> , [MAXFILES <i>number</i> , MEGABYTES <i>n</i> ]	Specifies the remote file to which the load data will be written. Oracle GoldenGate creates this file during the load.  <b>Note:</b> The size of an extract file cannot exceed 2GB.
<ul style="list-style-type: none"> <li>■ <i>path name</i> is the relative or fully qualified name of the file.</li> <li>■ MAXFILES creates a series of files that are aged as needed. Use if the file could exceed the operating system's file size limitations.</li> <li>■ MEGABYTES designates the size of each file.</li> </ul>	
TABLE <i>owner.table</i> ;	Specifies a source table or tables for initial data extraction.
<ul style="list-style-type: none"> <li>■ <i>owner</i> is the library or schema name.</li> <li>■ <i>table</i> is the name of the table or a group of tables defined with wildcards. To exclude tables from a wildcard specification, use the TABLEEXCLUDE parameter.</li> </ul>	

4. Enter any appropriate optional Extract parameters listed in the Oracle GoldenGate *Windows and UNIX Reference Guide*.
5. Save and close the parameter file.
6. On the target system, issue the following command to create an initial-load Replicat parameter file. This Replicat should have a different name from the Replicat group that applies the transactional data.

```
EDIT PARAMS initial-load Replicat name
```

7. Enter the parameters listed in [Table 5–2](#) in the order shown, starting a new line for each parameter statement.

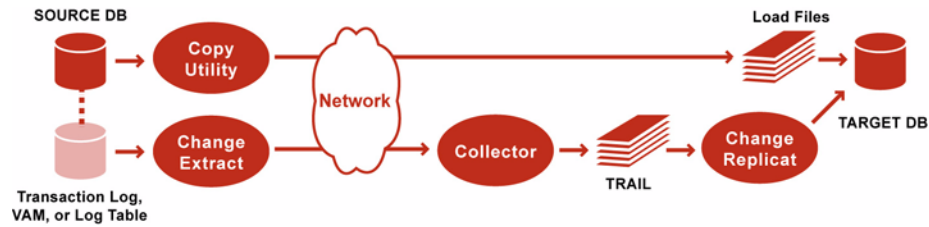
**Table 5–2 Initial-load Replicat parameters for loading data from file to Replicat**

Parameter	Description
SPECIALRUN	Implements the initial-load Replicat as a one-time run that does not use checkpoints.
END RUNTIME	Directs the initial-load Replicat to terminate when the load is finished.
TARGETDB <i>database</i> , USERID <i>user id</i> , PASSWORD <i>pw</i> , <i>algorithm</i> ENCRYPTKEY <i>keyname</i>	Specifies database connection information. <ul style="list-style-type: none"> <li>TARGETDB specifies the Data Source Name that is defined for the DB2 for i target database through the ODBC interface on the Windows or Linux system.</li> <li>USERID specifies the Replicat database user profile.</li> <li>PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command (see <a href="#">"Encrypting the Replicat password"</a> on page 4-9). Enter or paste the encrypted password after the PASSWORD keyword.</li> <li><i>algorithm</i> ENCRYPTKEY <i>keyname</i> specifies the encryption method and keyname that was specified in the ENCRYPT PASSWORD command.</li> </ul>
DECRYPTTRAIL BLOWFISH KEYNAME <i>keyname</i>	Decrypts the input trail. BLOWFISH is required because this is the algorithm that is supported to encrypt the file from DB2 for i.
EXTFILE <i>path name</i>   EXTTRAIL <i>path name</i>	Specifies the input extract file specified with the Extract parameter RMTFILE. <ul style="list-style-type: none"> <li><i>path name</i> is the relative or fully qualified name of the file or trail.</li> <li>Use EXTTRAIL only if you used the MAXFILES option of the RMTFILE parameter in the Extract parameter file.</li> </ul>
SOURCEDEFS <i>file name</i>   ASSUMETARGETDEFS	Specifies how to interpret data definitions. For more information about data definitions files, see <a href="#">Chapter 13, "Associating replicated data with metadata."</a> <ul style="list-style-type: none"> <li>Use SOURCEDEFS if the source and target tables have different definitions. Specify the relative or fully qualified name of the source-definitions file generated by the DEFGEN utility.</li> <li>Use ASSUMETARGETDEFS if the source and target tables have the same definitions.</li> </ul>
MAP <i>owner.table</i> , TARGET <i>owner.table</i> ;	Specifies a relationship between a source and target table or tables. <ul style="list-style-type: none"> <li><i>owner</i> is the schema name.</li> <li><i>table</i> is the name of a table or a wildcard definition for multiple tables. To exclude tables from a wildcard specification, use the MAPEXCLUDE parameter.</li> </ul>

8. Enter any appropriate optional Replicat parameters listed in the Oracle GoldenGate *Windows and UNIX Reference Guide*.
9. Save and close the file.

## 5.5.2 Configuring an initial load with a database utility

Figure 5–1



For an initial load between two DB2 for i source and target systems, you can use the DB2 for i system utilities to establish the target data. To do this, you save the file(s) that you want to load to the target by using the SAVOBJ or SAVLIB commands, and then you restore them on the target using the RSTOBJ or RSTLIB commands.

Another alternative is to use the IBM i commands CPYTOIMPF (Copy to Import File) and CPYFRMIMPF (Copy from Import File) to create files that can be used with the bulk load utilities of other databases. See the IBM i Information Center documentation for more details on "Copying between different systems."

In both cases, no special configuration of any Oracle GoldenGate initial-load processes is needed. You use the change-synchronization process groups that you configured in [Chapter 4](#). You start a change-synchronization Extract group to extract ongoing data changes while you are making the copy and loading it. When the copy is finished, you start the change-synchronization Replicat group to re-synchronize rows that were changed while the copy was being applied. From that point forward, both Extract and Replicat continue running to maintain data synchronization. See "[Adding change-capture and change-delivery processes](#)".

## 5.6 Adding change-capture and change-delivery processes

---

**Note:** Perform these steps at or close to the time that you are ready to start the initial load and change capture.

---

These steps establish the Oracle GoldenGate Extract, data pump, and Replicat processes that you configured in [Chapter 4](#). Collectively known as the "change-synchronization" processes, these are the processes that:

- capture and apply ongoing source changes while the load is being performed on the target
- reconcile any collisions that occur

---

**Note:** Perform these steps as close as possible to the time that you plan to start the initial load processes. You will start these processes during the initial load steps.

---

### 5.6.1 Add the primary Extract

These steps add the primary Extract that captures change data.

### 5.6.1.1 Understanding the primary Extract start point

When you add the primary Extract group, you establish an initial start position for data capture. This initial position can be a transaction boundary that is based on either of the following:

- a timestamp
- the end of the journal(s)
- a specific *system* sequence number
- a specific *journal* sequence number (per journal)

The options that are available to you assume a global start point and optional journal-specific start points.

- To position by a timestamp, at the end of the journals, or at a system sequence number, you will use the `ADD EXTRACT` command with the appropriate option. This command establishes a global start point for all journals and is a required first step.
- After issuing the `ADD EXTRACT` command, you can then optionally position any *given* journal at a specific journal sequence number by using the `ALTER EXTRACT` command with an appropriate journal option.

### 5.6.1.2 Establishing the required and optional Extract start points

These steps include the `ADD EXTRACT` and `ALTER EXTRACT` commands to enable you to establish your desired start points.

1. Run GGSCI.
2. Issue the `ADD EXTRACT` command to add the primary Extract group and establish the global start point.

```
ADD EXTRACT group name, TRANLOG
{
, BEGIN {NOW | yyyy-mm-dd[hh:mi:[ss[.cccccc]]]} |
, EOF |
, SEQNO seqno
}
```

Where:

- *group name* is the name of the primary Extract group that captures the transactional changes.
- TRANLOG specifies the journals as the data source.
- BEGIN specifies to begin capturing data as of a specific *time*. Select one of two options: NOW starts at the first record that is timestamped at the same time that BEGIN is issued. *yyyy-mm-dd[hh:mi:[ss[.cccccc]]]* starts at an explicit timestamp. Logs from this timestamp must be available.
- SEQNO *seqno* specifies to begin capturing data at, or just after, a system sequence number, which is a decimal number up to 20 digits in length.

#### Example 5–1 Timestamp start point

```
ADD EXTRACT finance, TRANLOG, BEGIN 2011-01-01 12:00:00.000000
```

#### Example 5–2 NOW start point

```
ADD EXTRACT finance, TRANLOG, BEGIN NOW
```

**Example 5-3 System sequence number start point**

```
ADD EXTRACT finance, TRANLOG, SEQNO 2952
```

3. (Optional) Issue the following command to alter any ADD EXTRACT start position to set the start position for a specific journal in the same Extract configuration. A *specific* journal position set with ALTER EXTRACT does not affect any global position that was previously set with ADD EXTRACT or ALTER EXTRACT; however a *global* position set with ALTER EXTRACT overrides any specific journal positions that were previously set in the same Extract configuration.

```
ALTER EXTRACT group name,
{
ALTER EXTRACT {BEGIN {NOW | yyyy-mm-dd [hh:mi:[ss[.cccccc]]} [JOURNAL journal_
library/journal_name [[JRNRCV receiver_library/receiver_name]] |
, EOF [JOURNAL journal_library/journal_name [[JRNRCV receiver_library/receiver_
name]] |
, SEQNO seqno [JOURNAL journal_library/journal_name [[JRNRCV receiver_
library/receiver_name]]
}
```

---

**Note:** SEQNO, when used with a journal in ALTER EXTRACT, is the journal sequence number that is relative to that specific journal, not the system sequence number that is global across journals.

---

**Example 5-4 Journal start point**

```
ALTER EXTRACT finance, SEQNO 1234 JOURNAL accts/acctsjrn
```

**Example 5-5 Journal and receiver start point**

```
ALTER EXTRACT finance, SEQNO 1234 JOURNAL accts/acctsjrn JRNRCV accts/jrnrcv0005
```

## 5.6.2 Add the local trail

This step adds the local trail to which the primary Extract writes captured data.

In GGSCI on the source system, issue the ADD EXTTRAIL command:

```
ADD EXTTRAIL pathname, EXTRACT group name
```

Where:

- EXTTRAIL specifies that the trail is to be created on the local system.
- *pathname* is the relative or fully qualified name of the trail, including the two-character name.
- EXTRACT *group name* is the name of the primary Extract group.

**Example 5-6**

```
ADD EXTTRAIL /ggs/dirdat/lt, EXTRACT finance
```

## 5.6.3 Add the data pump Extract group

This step adds the data pump that reads the local trail and sends the data to the target.

In GGSCI on the source system, issue the ADD EXTRACT command.

```
ADD EXTRACT group name, EXTTRAILSOURCE trail name
```

Where:

- *group name* is the name of the data-pump Extract group.
- EXTTRAILSOURCE *trail name* is the relative or fully qualified name of the local trail.

**Example 5–7**

```
ADD EXTRACT financep, EXTTRAILSOURCE c:\ggs\dirdat\lt
```

## 5.6.4 Add the remote trail

This step adds the remote trail. Although it is read by Replicat, this trail must be associated with the data pump, so it must be added on the source system, not the target.

In GGSCI on the source system, issue the following command:

```
ADD RMTTRAIL pathname, EXTRACT group name
```

Where:

- RMTTRAIL specifies that the trail is to be created on the target system, and *pathname* is the relative or fully qualified name of the trail, including the two-character name.
- EXTRACT *group name* is the name of the data-pump Extract group.

**Example 5–8**

```
ADD RMTTRAIL /ggs/dirdat/rt, EXTRACT financep
```

## 5.6.5 Add the Replicat group

These steps add the Replicat group that reads the remote trail (which gets created automatically on the target) and applies the data changes to the target Oracle database.

1. Run GGSCI on the target system.
2. Issue the ADD REPLICAT command.

```
ADD REPLICAT group name, EXTTRAIL pathname
```

Where:

- *group name* is the name of the Replicat group.
- EXTTRAIL *pathname* is the relative or fully qualified name of the remote trail, including the two-character name.

**Example 5–9**

```
ADD REPLICAT financer, EXTTRAIL c:\ggs\dirdat\rt
```

## 5.7 Performing the target instantiation

This procedure instantiates the target tables while Oracle GoldenGate captures ongoing transactional changes on the source and stores them until they can be applied

on the target. By the time you perform the instantiation of the target tables, the entire Oracle GoldenGate environment should be configured for change capture and delivery, as should the initial-load processes if using Oracle GoldenGate as an initial-load utility.

### 5.7.1 To perform instantiation from file to Replicat

1. Make certain that you have addressed the requirements in ["Satisfying prerequisites for instantiation"](#) on page 5-2.
2. On the source and target systems, run GGSCI and start the Manager process.

```
START MANAGER
```

3. On the source system, start the primary and data pump Extract groups to start change extraction.

```
START EXTRACT primary Extract group name
START EXTRACT data pump Extract group name
```

4. From the directory where Oracle GoldenGate is installed on the source system, start the initial-load Extract as follows:

```
$ /GGS directory/extract paramfile dirprm/initial-load Extract name.prm
reportfile path name
```

Where: *initial-load Extract name* is the name of the initial-load Extract that you used when creating the parameter file, and *path name* is the relative or fully qualified name of the Extract report file (by default the *dirrpt* sub-directory of the Oracle GoldenGate installation directory).

5. Verify the progress and results of the initial extraction by viewing the Extract report file using the operating system's standard method for viewing files.
6. Wait until the initial extraction is finished.
7. On the target system, start the initial-load Replicat.

```
$ /GGS directory/replicat paramfile dirprm/initial-load Replicat name.prm
reportfile path name
```

Where: *initial-load Replicat name* is the name of the initial-load Replicat that you used when creating the parameter file, and *path name* is the relative or fully qualified name of the Replicat report file (by default the *dirrpt* sub-directory of the Oracle GoldenGate installation directory).

8. When the initial-load Replicat is finished running, verify the results by viewing the Replicat report file using the operating system's standard method for viewing files.
9. On the target system, start change replication.

```
START REPLICAT Replicat group name
```

10. On the target system, issue the following command to verify the status of change replication.

```
INFO REPLICAT Replicat group name
```

11. Continue to issue the `INFO REPLICAT` command until you have verified that Replicat posted all of the change data that was generated during the initial load. For example, if the initial-load Extract stopped at 12:05, make sure Replicat posted data up to that point.

12. On the target system, issue the following command to turn off the `HANDLECOLLISIONS` parameter and disable the initial-load error handling.  

```
SEND REPLICAT Replicat group name, NOHANDLECOLLISIONS
```
13. On the target system, edit the Replicat parameter file to remove the `HANDLECOLLISIONS` parameter. This prevents `HANDLECOLLISIONS` from being enabled again the next time Replicat starts.

---

---

**Caution:** Do not use the `VIEW PARAMS` or `EDIT PARAMS` command to view or edit an existing parameter file that is in a character set other than that of the local operating system (such as one where the `CHARSET` option was used to specify a different character set). View the parameter file from outside GGSCI if this is the case; otherwise, the contents may become corrupted.

---

---

14. Save and close the parameter file.

From this point forward, Oracle GoldenGate continues to synchronize data changes.

## 5.7.2 To perform instantiation with a database utility

1. Make certain that you have addressed the requirements in ["Satisfying prerequisites for instantiation"](#) on page 5-2.
2. On the source and target systems, run GGSCI and start the Manager process.  

```
START MANAGER
```
3. On the source system, start the primary and data pump Extract groups to start change extraction.  

```
START EXTRACT primary Extract group name  
START EXTRACT data pump Extract group name
```
4. On the source system, start making the copy.
5. Wait until the copy is finished and record the time of completion.
6. View the Replicat parameter file to make certain that the `HANDLECOLLISIONS` parameter is listed. If not, edit the file and add the parameter to the file.

```
EDIT PARAMS Replicat group name
```

---

---

**Caution:** Do not use the `VIEW PARAMS` or `EDIT PARAMS` command to view or edit an existing parameter file that is in a character set other than that of the local operating system (such as one where the `CHARSET` option was used to specify a different character set). View the parameter file from outside GGSCI if this is the case; otherwise, the contents may become corrupted.

---

---

7. On the target system, start change replication.  

```
START REPLICAT Replicat group name
```
8. On the target system, issue the following command to verify the status of change replication.

---

```
INFO REPLICAT Replicat group name
```

9. Continue to issue the `INFO REPLICAT` command until you have verified that change replication has posted all of the change data that was generated during the initial load. Reference the time of completion that you recorded. For example, if the copy stopped at 12:05, make sure change replication has posted data up to that point.

10. On the target system, issue the following command to turn off the `HANDLECOLLISIONS` parameter and disable the initial-load error handling.

```
SEND REPLICAT Replicat group name, NOHANDLECOLLISIONS
```

11. On the target system, edit the Replicat parameter file to remove the `HANDLECOLLISIONS` parameter. This prevents `HANDLECOLLISIONS` from being enabled again the next time Replicat starts.

---

**Caution:** Do not use the `VIEW PARAMS` or `EDIT PARAMS` command to view or edit an existing parameter file that is in a character set other than that of the local operating system (such as one where the `CHARSET` option was used to specify a different character set). View the parameter file from outside GGSCI if this is the case; otherwise, the contents may become corrupted.

---

12. Save and close the parameter file.

From this point forward, Oracle GoldenGate continues to synchronize data changes.

## 5.8 Monitoring processing after the instantiation

After the target is instantiated and replication is in effect, you should view the status, lag, and overall health of the replication environment to ensure that processes are running properly, that there are no warnings in the Oracle GoldenGate error log, and that lag is at an acceptable level. You can view Oracle GoldenGate processes from:

- *GGSCI:* See the Oracle GoldenGate *Windows and UNIX Administrator's Guide* for information about monitoring processes.
- *Oracle GoldenGate Monitor:* See the administration documentation and online help for that product. Oracle GoldenGate Monitor provides a graphical-based monitoring environment for all of your Oracle GoldenGate instances.

You also should verify that capture and delivery is being performed for all of the tables in the Oracle GoldenGate configuration, and that the source and target data are synchronized. You can use the Oracle GoldenGate Veridata product for this purpose.

## 5.9 Backing up the Oracle GoldenGate environment

After you start Oracle GoldenGate processing, an effective backup routine is critical to preserving the state of processing in the event of a failure. Unless the Oracle GoldenGate working files can be restored, the entire replication environment must be re-instantiated, complete with new initial loads.

As a best practice, include the entire Oracle GoldenGate home installation in your backup routines. This directory contains critical sub-directories, files and programs. The most critical working files in this directory consume the vast majority of backup

space; therefore it makes sense just to back up the entire installation directory for fast, simple recovery.

## 5.10 Positioning Extract after startup

You may at some point, over the life of an Extract run, need to set the position of Extract in the data stream manually. To reposition Extract, use the `ALTER EXTRACT` command in GGSCI. To help you identify any given Extract read position, the `INFO EXTRACT` command shows the positions for each journal in an Extract configuration, including the journal receiver information. For more information about these commands, see the Oracle GoldenGate *Windows and UNIX Reference Guide*

---

---

**Note:** Because the journals can have a transaction split among them, if a given journal is independently repositioned far into the past, the resulting latency from reprocessing the entries may cause the already-read journals to stall until the reading of the latent journal catches up.

---

---

---



---

## Uninstalling Oracle GoldenGate

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

### 6.1 Uninstalling Oracle GoldenGate from a source DB2 for i system

1. Run the command shell of the operating system.
2. (Suggested) Log on to the operating system as the system administrator or as a user with permission to issue Oracle GoldenGate commands and delete files and directories from the operating system.
3. Change directories to the Oracle GoldenGate installation directory.
4. Run GGSCI.
5. Stop all Oracle GoldenGate processes.

```
STOP ER *
```

6. Stop the Manager process.

```
STOP MANAGER
```

7. Log into the database with the DBLOGIN command.

```
DBLOGIN SOURCEDB database, USERID db_user [, PASSWORD pw [encryption options]]
```

---



---

**Note:** Only BLOWFISH encryption is supported for DB2 for i systems.

---



---

8. Run the DELETE TRANDATA command to end the journaling of tables.

```
DELETE TRANDATA { schema.table [JOURNAL library/journal] |  
library/file [JOURNAL library/journal] }
```

9. From an IBM 5250 terminal emulator, find the Oracle GoldenGate installation library. This library can be seen as the name in the oggprcjrnl.srvpgm soft-link after qsys.lib. In the following example, the installation library is OGGLIB:

```
oggprcjrnl.srvpgm->/qsys.lib/qsys.ogglib/oggprcjrnl.srvpgm
```

10. Delete the Oracle GoldenGate installation library. Specify "I" (ignore) for any prompts about unsaved journal receivers.

`clrlib library`

`dltlib library`

11. Remove the Oracle GoldenGate files by removing the installation directory.

## 6.2 Uninstalling Oracle GoldenGate from a Linux remote system

Follow these instructions to remove Oracle GoldenGate from a remote Linux machine where Replicat is installed.

1. Run the command shell of the operating system.
2. (Suggested) Log on to the operating system as the system administrator or as a user with permission to issue Oracle GoldenGate commands and delete files and directories from the operating system.
3. Change directories to the Oracle GoldenGate installation directory.
4. Run GGSCI.
5. Stop all Oracle GoldenGate processes.

```
STOP ER *
```

6. Stop the Manager process.

```
STOP MANAGER
```

7. Log into the database with the DBLOGIN command.

```
DBLOGIN SOURCEDB database, USERID db_user [, PASSWORD pw [encryption options]]
```

---

---

**Note:** Only BLOWFISH encryption is supported for DB2 for i systems.

---

---

8. Remove the Replicat checkpoint table by running the `DELETE CHECKPOINTTABLE` command.

```
DELETE CHECKPOINTTABLE owner.table
```

9. Remove the Oracle GoldenGate files by removing the installation directory.

## 6.3 Removing Oracle GoldenGate from a Windows remote system

Follow these instructions to remove Oracle GoldenGate from a remote Windows machine where Replicat is installed.

### 6.3.1 Removing Oracle GoldenGate from a Windows cluster

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

6. Run the INSTALL utility using the following syntax.

```
install deleteevents deleteservice
```

This command stops Oracle GoldenGate events from being reported to the Windows Event Manager and removes the Manager service.

7. Delete the CATEGORY.DLL and GGSMMSG.DLL files from the Windows SYSTEM32 folder.
8. Move the cluster group to the next node in the cluster, and repeat from 4.
9. Follow the instructions in [Removing Oracle GoldenGate from a remote Windows system](#).

## 6.3.2 Removing Oracle GoldenGate from a remote Windows system

### On all systems:

1. (Suggested) Log on to the operating system as the system administrator or as a user with permission to issue Oracle GoldenGate commands and to delete files and directories from the operating system.
2. From the Oracle GoldenGate installation folder, run GGSCI.
3. Stop all Oracle GoldenGate processes.

```
STOP ER *
```

4. Stop the Manager process.

```
STOP MANAGER
```

---



---

**Note:** Skip Steps 5 through 8 if you already performed them when removing Oracle GoldenGate from a Windows cluster

---



---

5. Click **Start**, then **Run**, and type `cmd` in the Run dialog box to open the command console.
6. Change directories to the Oracle GoldenGate installation directory.
7. Run the INSTALL utility using the following syntax.

```
install deleteevents deleteservice
```

This command stops Oracle GoldenGate events from being reported to the Windows Event Manager and removes the Manager service.

8. Delete the CATEGORY.DLL and GGSMMSG.DLL files from the Windows SYSTEM32 folder.
9. Log into the database with the DBLOGIN command.

```
DBLOGIN SOURCEDB database, USERID db_user [, PASSWORD pw [encryption options]]
```

---



---

**Note:** Only BLOWFISH encryption is supported for DB2 for i systems.

---



---

10. Remove the Replicat checkpoint table by running the DELETE CHECKPOINTTABLE command.

```
DELETE CHECKPOINTTABLE owner.table
```

11. Remove the Oracle GoldenGate files by removing the installation directory.

---



---

## Oracle GoldenGate installed components

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

### A.1 Oracle Goldengate Programs And Utilities

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

---



---

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

---



---

**Table A-1 Oracle GoldenGate installed programs and utilities**

Program	Description
convchk	Converts checkpoint files to a newer version.
ddlgen	Generates target database table definitions based on source database DDL. Used primarily on the NonStop platform.
defgen	Generates data definitions and is referenced by Oracle GoldenGate processes when source and target tables have dissimilar definitions.
emscInt	Sends event messages created by Collector and Replicat on Windows or UNIX systems to EMS on NonStop systems.
extract	Performs capture from database tables or transaction logs or receives transaction data from a vendor access module.
ggminstall	Oracle GoldenGate installation script for the SQL/MX database.
ggsci	User interface to Oracle GoldenGate for issuing commands and managing parameter files.
ggsMgr.jcl	Start the Oracle GoldenGate Manager process from a batch job or the operator console on a z/OS system. Installed to support DB2 z/OS databases.
ggsMgr.proc	
ggsMgrst.jcl	
ggsMgrst.proc	

**Table A-1 (Cont.) Oracle GoldenGate installed programs and utilities**

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

## A.2 Oracle Goldengate Subdirectories

This Section describes the subdirectories of the Oracle Goldengate installation directory and their contents.

---



---

**Note:** Some directories may not exist in all installations.

---



---

**Table A-2 Oracle GoldenGate installed subdirectories**

Directory	Description
br	Contains the checkpoint files for the bounded recover feature.
cfg	Contains the property and XML files that are used to configure Oracle GoldenGate Monitor.
dirdb	Contains the datastore that is used to persist information that is gathered from an Oracle GoldenGate instance for use by the Oracle GoldenGate Monitor application or within Oracle Enterprise Manager.

**Table A-2 (Cont.) Oracle GoldenGate installed subdirectories**

Directory	Description
dirchk	<p>Contains the checkpoint files created by Extract and Replicat processes, which store current read and write positions to support data accuracy and fault tolerance. Written in internal Oracle GoldenGate format.</p> <p>File name format is <i>group_name+sequence_number.ext</i> where <i>sequence_number</i> is a sequential number appended to aged files and <i>ext</i> is either <i>cpe</i> for Extract checkpoint files or <i>cpr</i> for Replicat checkpoint files.</p> <p>Do not edit these files.</p> <p>Examples:</p> <p>ext1.cpe</p> <p>repl.cpr</p>
dirdat	<p>The default location for Oracle GoldenGate trail files and extract files that are created by Extract processes to store extracted data for further processing by the Replicat process or another application or utility. Written in internal Oracle GoldenGate format.</p> <p>File name format is a user-defined two-character prefix followed by either a six-digit sequence number (trail files) or the user-defined name of the associated Extract process group (extract files).</p> <p>Do not edit these files.</p> <p>Examples:</p> <p>rt000001</p> <p>finance</p>
dirdef	<p>The default location for data definitions files created by the DEFGEN utility to contain source or target data definitions used in a heterogeneous synchronization environment. Written in external ASCII. File name format is a user-defined name specified in the DEFGEN parameter file.</p> <p>These files may be edited to add definitions for newly created tables. If you are unsure of how to edit a definitions file, contact Oracle GoldenGate technical support.</p> <p>Example:</p> <p>defs.dat</p>
dirjar	<p>Contains the Java executable files that support Oracle GoldenGate Monitor.</p>
dirout	<p>This directory is not used any more.</p>
dirpcs	<p>Default location for status files. File name format is <i>group.extension</i> where <i>group</i> is the name of the group and <i>extension</i> is either <i>pce</i> (Extract), <i>pcr</i> (Replicat), or <i>pcm</i> (Manager).</p> <p>These files are only created while a process is running. The file shows the program name, the process name, the port number, and the process ID.</p> <p>Do not edit these files.</p> <p>Examples:</p> <p>mgr.pcm</p> <p>ext.pce</p>

**Table A–2 (Cont.) Oracle GoldenGate installed subdirectories**

Directory	Description
dirprm	<p>The default location for Oracle GoldenGate parameter files created by Oracle GoldenGate users to store run-time parameters for Oracle GoldenGate process groups or utilities. Written in external ASCII format. File name format is <i>group name/user-defined name.prm</i> or <i>mgr.prm</i>.</p> <p>These files may be edited to change Oracle GoldenGate parameter values after stopping the process. They can be edited directly from a text editor or by using the <code>EDIT PARAMS</code> command in GGSCI.</p> <p>Examples:</p> <p><code>defgen.prm</code></p> <p><code>finance.prm</code></p>
dirrec	Not used by Oracle GoldenGate.
dirrpt	<p>The default location for process report files created by Extract, Replicat, and Manager processes to report statistical information relating to a processing run. Written in external ASCII format.</p> <p>File name format is <i>group name+sequence number.rpt</i> where <i>sequence number</i> is a sequential number appended to aged files.</p> <p>Do not edit these files.</p> <p>Examples:</p> <p><code>fin2.rpt</code></p> <p><code>mgr4.rpt</code></p>
dirsql	Used by the <code>triggen</code> utility to store SQL scripts before <code>triggen</code> was deprecated. Currently used to store training scripts and any user-created SQL scripts that support Oracle GoldenGate.
dirtmp	The default location for storing transaction data when the size exceeds the memory size that is allocated for the cache manager. Do not edit these files.
dirwlt	Contains the Oracle Wallet that supports Oracle GoldenGate Monitor. This directory is not installed until you run the utility that creates the wallet.
UserExitExamples	Contains sample files to help with the creation of user exits.

### A.3 Other Oracle GoldenGate files

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

---



---

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

---



---

**Table A–3 Other Oracle GoldenGate installed files**

Component	Description
<code>bcpfmt.tpl</code>	Template for use with Replicat when creating a run file for the Microsoft BCP/DTS bulk-load utility.
<code>bcrypt.txt</code>	Blowfish encryption software license agreement.

**Table A-3 (Cont.) Other Oracle GoldenGate installed files**

<b>Component</b>	<b>Description</b>
cagent.dll	Contains the Windows dynamic link library for the Oracle GoldenGate Monitor C sub-agent.
category.dll	Windows dynamic link library used by the INSTALL utility.
chkpt_db_create.sql	Script that creates a checkpoint table in the local database. A different script is installed for each database type.
db2cntl.tpl	Template for use with Replicat when creating a control file for the IBM LOADUTIL bulk-load utility.
ddl_access.tpl	Template used by the DDLGEN utility to convert source DDL to Microsoft Access DDL.
ddl_cleartrace.sql	Script that removes the DDL trace file. (Oracle installations)
ddl_db2.tpl	Template used by the DDLGEN utility to convert source DDL to DB2 DDL (Linux, UNIX, Windows).
ddl_db2_os390.tpl	Template used by the DDLGEN utility to convert source DDL to DB2 DDL (z/OS systems).
ddl_ddl2file.sql	Script that saves DDL from the marker table to a file.
ddl_disable.sql	Script that disables the Oracle GoldenGate DDL trigger. (Oracle installations)
ddl_enable.sql	Script that enables the Oracle GoldenGate DDL trigger. (Oracle installations)
ddl_filter.sql	Script that supports filtering of DDL by Oracle GoldenGate. This script runs programmatically; do not run it manually.
ddl_informix.tpl	Template used by the DDLGEN utility to convert source DDL to Informix DDL.
ddl_mss.tpl	Template used by the DDLGEN utility to convert source DDL to SQL Server DDL.
ddl_mysql.tpl	Template used by the DDLGEN utility to convert source DDL to MySQL DDL.
ddl_nopurgeRecyclebin.sql	Empty script file for use by Oracle GoldenGate support staff.
ddl_nssql.tpl	Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.
ddl_ora9.sql	Scripts that run programmatically as part of Oracle GoldenGate DDL support; do not run these scripts.
ddl_ora10.sql	
ddl_ora11.sql	
ddl_ora10upCommon.sql	
ddl_oracle.tpl	Template used by the DDLGEN utility to convert source DDL to Oracle DDL.
ddl_pin.sql	Script that pins DDL tracing, the DDL package, and the DDL trigger for performance improvements. (Oracle installations)
ddl_purgeRecyclebin.sql	Script that purges the Oracle recyclebin in support of the DDL replication feature.
ddl_remove.sql	Script that removes the DDL extraction trigger and package. (Oracle installations)
ddl_session.sql	Supports the installation of the Oracle DDL objects. This script runs programmatically; do not run it manually.
ddl_session1.sql	

**Table A-3 (Cont.) Other Oracle GoldenGate installed files**

<b>Component</b>	<b>Description</b>
ddl_setup.sql	Script that installs the Oracle GoldenGate DDL extraction and replication objects. (Oracle installations)
ddl_sqlmx.tpl	Template used by the DDLGEN utility to convert Tandem Enscribe DDL to NonStop SQL/MX DDL.
ddl_status.sql	Script that verifies whether or not each object created by the Oracle GoldenGate DDL support feature exists and is functioning properly. (Oracle installations)
ddl_staymetadata_off.sql ddl_staymetadata_on.sql	Scripts that control whether the Oracle DDL trigger collects metadata. This script runs programmatically; do not run it manually.
ddl_sybase.tpl	Template used by the DDLGEN utility to convert source DDL to Sybase DDL.
ddl_tandem.tpl	Template used by the DDLGEN utility to convert source DDL to NonStop SQL DDL.
ddl_trace_off.sql ddl_trace_on.sql	Scripts that control whether DDL tracing is on or off.
ddl_tracelevel.sql	Script that sets the level of tracing for the DDL support feature. (Oracle installations)
debug files	Debug text files that may be present if tracing was turned on.
demo_db_scriptname.sql demo_more_db_scriptname.sql	Scripts that create and populate demonstration tables for use with tutorials and basic testing.
.dmp files	Dump files created by Oracle GoldenGate processes for tracing purposes.
ENCKEYS	User-created file that stores encryption keys. Written in external ASCII format.
exitdemo.c	User exit example.
exitdemo_utf16.c	User exit example that demonstrates how to use UTF16 encoded data in the callback structures for information exchanged between the user exit and the process.
freeBSD.txt	License agreement for FreeBSD.
ggmessage.dat	Data file that contains error, informational, and warning messages that are returned by the Oracle GoldenGate processes. The version of this file is checked upon process startup and must be identical to that of the process in order for the process to operate.
ggserr.log	File that logs processing events, messages, errors, and warnings generated by Oracle GoldenGate.
ggsmsg.dll	Windows dynamic link library used by the install program.
GLOBALS	User-created file that stores parameters applying to the Oracle GoldenGate instance as a whole.
help.txt	Help file for the GGSCI command interface.
icudt38.dll icuin38.dll icuuc38.dll	Windows shared libraries for International Components for Unicode.

**Table A-3 (Cont.) Other Oracle GoldenGate installed files**

<b>Component</b>	<b>Description</b>
jagent.bat	Windows batch file for the Java Agent for Oracle GoldenGate Monitor.
jagent.log	Log files for the Oracle GoldenGate Monitor Agent.
jagentjni.log	
jagent.sh	UNIX shell script for the Java Agent for Oracle GoldenGate Monitor
LGPL.txt	Lesser General Public License statement. Applies to free libraries from the Free Software Foundation.
libodbc.so	ODBC file for Ingres 2.6 on Unix.
libodbc.txt	License agreement for libodbc.so.
libxml2.dll	Windows dynamic link library containing the XML library for the Oracle GoldenGate XML procedures.
libxml2.txt	License agreement for libxml2.dll.
marker.hist	File created by Replicat if markers were passed from a NonStop source system.
marker_remove.sql	Script that removes the DDL marker table. (Oracle installations)
marker_setup.sql	Script that installs the Oracle GoldenGate DDL marker table. (Oracle installations)
marker_status.sql	Script that confirms successful installation of the DDL marker table. (Oracle installations)
notices.txt	Third-party software license file.
odbcinst.ini	Ingres 2.6 on Unix ODBC configuration file.
oggrcjrnsrvpgm	Link to the OGGPRCJRN service program on IBM i installations
params.sql	Script that contains configurable parameters for DDL support. (Oracle installations)
pthread-win32.txt	License agreement for pthread-VC.dll.
pthread-VC.dll	POSIX threads library for Microsoft Windows.
prvtclkm.plb	Supports the replication of Oracle encrypted data.
pw_agent_util.bat	Script files that support the Oracle GoldenGate Monitor Agent.
pw_agent_util.sh	
role_setup.sql	Script that creates the database role necessary for Oracle GoldenGate DDL support. (Oracle installations)
sampleodbc.ini	Sample ODBC file for Ingres 2.6 on UNIX.
sqlldr.tpl	Template for use with Replicat when creating a control file for the Oracle SQL*Loader bulk-load utility.
start.prm	z/OS parmlib members to start and stop the Manager process.
stop.prm	
startmgr	z/OS Unix System Services scripts to start the Manager process from GGSCI.
stopmgr	
startmgrcom	z/OS system input command for the Manager process.
stopmgrcom	

**Table A-3 (Cont.) Other Oracle GoldenGate installed files**

Component	Description
tcperrs	File containing user-defined instructions for responding to TCP/IP errors.
usrdecs.h	Include file for user exit API.
xerces-c_2_8.dll	Apache XML parser library.
zlib.txt	License agreement for zlib compression library.

## A.4 Oracle GoldenGate checkpoint table

When database checkpoints are being used, Oracle GoldenGate creates a checkpoint table with a user-defined name in the database upon execution of the `ADD CHECKPOINTTABLE` command, or a user can create the table by using the `chkpt_db_create.sql` script (where *db* is an abbreviation of the type of database that the script supports).

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

**Table A-4 Checkpoint table definition**

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