

Oracle® Fusion Middleware

Installing and Configuring Oracle GoldenGate for DB2 for i
12c (12.2.0.1)

E66362-03

February 2017

Documentation for installers and system administrators that describes how to install and configure Oracle GoldenGate for the DB2 for i Database.

Copyright © 2013, 2017, 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 END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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 failsafe, 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 about 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 unless otherwise set forth in an applicable agreement between you and Oracle. 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, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	vii
----------------------	------------

1 System Requirements and Preinstallation Instructions

1.1	Overview of Oracle GoldenGate for IBM DB2 for i	1-1
1.2	Verifying Certification and System Requirements	1-2
1.3	Operating System Requirements	1-2
1.3.1	General Requirements.....	1-2
1.3.2	Memory Requirements	1-3
1.3.3	Disk Requirements	1-4
1.3.4	Network Requirements.....	1-4
1.3.5	Oracle GoldenGate Security Privileges	1-5
1.4	Supported DB2 for i Data Types.....	1-6
1.5	Non-Supported DB2 for i Data Types.....	1-6
1.6	Supported Objects and Operations for DB2 for i	1-6
1.7	Non-Supported Objects and Operations for DB2 for i	1-7
1.8	Oracle GoldenGate Parameters Not Supported for DB2 for i	1-7
1.9	Supported Object Naming Conventions	1-8
1.10	Supported Character Sets	1-8

2 Installing Oracle GoldenGate

2.1	Installation Overview	2-1
2.2	Understanding and Obtaining the Oracle GoldenGate Distribution.....	2-1
2.3	Setting Library Paths for Dynamic Builds on DB2 for i	2-2
2.4	Installing Oracle GoldenGate on DB2 for i	2-3
2.5	Installing Oracle GoldenGate on Linux	2-5
2.6	Installing Oracle GoldenGate on Windows	2-5
2.6.1	Installing Oracle GoldenGate into a Windows Cluster	2-5
2.6.2	Installing the Oracle GoldenGate Files	2-6
2.6.3	Specifying a Custom Manager Name	2-6
2.6.4	Installing Manager as a Windows Service	2-7

3 Preparing the System for Oracle GoldenGate

3.1	Preparing the Journals for Data Capture by Extract.....	3-1
3.1.1	Allocating Journals to an Extract Group	3-1
3.1.2	Setting Journal Parameters	3-1

3.1.3	Deleting Old Journal Receivers.....	3-2
3.2	Specifying Object Names	3-2
3.3	Preparing Tables for Processing	3-3
3.3.1	Assigning Row Identifiers	3-3
3.3.2	Preventing Key Changes	3-4
3.3.3	Disabling Constraints on the Target	3-4
3.3.4	Enabling Change Capture	3-5
3.3.5	Maintaining Materialized Query Tables	3-6
3.3.6	Specifying the Oracle GoldenGate Library	3-6
3.4	Adjusting the System Clock	3-6
3.5	Configuring the ODBC Driver	3-6
3.5.1	Configuring ODBC on Linux	3-6
3.5.2	Configuring ODBC on Windows	3-10

4 Configuring Oracle GoldenGate for DB2 for i

4.1	What to Expect from this Procedure	4-1
4.2	Getting Started with Oracle GoldenGate	4-2
4.3	Creating the Oracle GoldenGate Instance	4-2
4.4	Creating a GLOBALS File	4-2
4.5	Creating a Data Definitions File.....	4-2
4.6	Encrypting the Extract and Replicat Passwords	4-3
4.7	Configuring Extract for Change Capture from DB2 for i.....	4-3
4.7.1	Configuring the Primary Extract	4-3
4.7.2	Configuring the Data Pump.....	4-4
4.8	Configuring Replicat for Change Delivery to DB2 for i.....	4-5
4.8.1	Creating a Checkpoint Table.....	4-6
4.8.2	Configuring Replicat	4-6
4.9	Next Steps in the Deployment	4-7
4.10	When to Start Replicating Transactional Changes.....	4-8
4.10.1	Starting Extract During Instantiation.....	4-8
4.10.2	Changing the Position of Extract to a Later Time	4-8
4.11	Testing Your Configuration.....	4-8

5 Instantiating and Starting Oracle GoldenGate Replication

5.1	About the Instantiation Process	5-1
5.2	Overview of Basic Oracle GoldenGate Instantiation Steps	5-2
5.3	Satisfying Prerequisites for Instantiation	5-2
5.3.1	Configure Change Capture and Delivery	5-2
5.3.2	Add Collision Handling	5-2
5.3.3	Prepare the Target Tables	5-3
5.4	Making the Instantiation Procedure More Efficient	5-3
5.4.1	Share Parameters Between Process Groups.....	5-3
5.4.2	Use Parallel Processes	5-3
5.5	Configuring the Initial Load.....	5-3
5.5.1	Configuring an Initial Load from File to Replicat.....	5-4
5.5.2	Configuring an initial load with a database utility.....	5-7
5.6	Adding Change-Capture and Change-Delivery processes	5-7

5.6.1	Add the Primary Extract.....	5-8
5.6.2	Add the Local Trail.....	5-9
5.6.3	Add the Data Pump Extract Group	5-10
5.6.4	Add the Remote Trail.....	5-10
5.6.5	Add the Replicat Group.....	5-10
5.7	Performing the Target Instantiation.....	5-11
5.7.1	To Perform Instantiation from File to Replicat.....	5-11
5.7.2	To Perform Instantiation with a Database Utility	5-12
5.8	Monitoring Processing after the Instantiation	5-13
5.9	Backing up Your Oracle GoldenGate Environment	5-14
5.10	Positioning Extract After Startup	5-14

6 Using Remote Journal

6.1	Preparing to Use Remote Journals	6-1
6.2	Adding a Remote Journal	6-2
6.2.1	What Happens During Add Remote Journal Processing?	6-3
6.2.2	Guidelines For Adding a Remote Journal.....	6-3

7 Uninstalling Oracle GoldenGate

7.1	Uninstalling Oracle GoldenGate from a Source DB2 for i System	7-1
7.2	Uninstalling Oracle GoldenGate from a Linux Remote System	7-2
7.3	Removing Oracle GoldenGate from a Windows Remote System	7-2
7.3.1	Removing Oracle GoldenGate from a Windows Cluster	7-2
7.3.2	Removing Oracle GoldenGate from a Remote Windows System	7-3

A Oracle GoldenGate Installed Components

A.1	Oracle GoldenGate Programs and Utilities.....	A-1
A.2	Oracle GoldenGate Subdirectories	A-2
A.3	Other Oracle GoldenGate Files	A-4
A.4	Oracle GoldenGate Checkpoint Table	A-7
A.5	Oracle GoldenGate DB2 for i Objects.....	A-7

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 Oracle GoldenGate configuration that is tailored to the DB2 for i environment. 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. Where needed, it points you to other documentation where you can find additional information to expand the configuration to suit your needs.

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 that have purchased support have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

The Oracle GoldenGate documentation set includes the following components:

Windows, UNIX, and Linux Platforms

- *Installing and Configuring Oracle GoldenGate for DB2 for i*

- *Installing and Configuring Oracle GoldenGate for DB2 LUW*
- *Installing and Configuring Oracle GoldenGate for DB2 z/OS*
- *Installing and Configuring Oracle GoldenGate for Informix*
- *Installing and Configuring Oracle GoldenGate for MySQL*
- *Installing and Configuring Oracle GoldenGate for NonStop SQL/MX*
- *Installing and Configuring Oracle GoldenGate for SQL Server*
- *Installing and Configuring Oracle GoldenGate for Oracle TimesTen*
- *Installing and Configuring Oracle GoldenGate for Oracle Database*
- *Installing and Configuring Oracle GoldenGate for Sybase*
- *Installing and Configuring Oracle GoldenGate for Teradata*
- *Administering Oracle GoldenGate for Windows and UNIX*
- *Reference for Oracle GoldenGate for Windows and UNIX*
- *Logdump Reference for Oracle GoldenGate*
- *Upgrading Oracle GoldenGate for Windows and UNIX*
- *Error Messages Reference for Oracle GoldenGate for Windows and UNIX*

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, such as "From the File menu, select Save ." Boldface also is used for terms defined in text or in the glossary.
<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.

This chapter contains the following sections:

- [Section 1.1, "Overview of Oracle GoldenGate for IBM DB2 for i"](#)
- [Section 1.2, "Verifying Certification and System Requirements"](#)
- [Section 1.3, "Operating System Requirements"](#)
- [Section 1.4, "Supported DB2 for i Data Types"](#)
- [Section 1.5, "Non-Supported DB2 for i Data Types"](#)
- [Section 1.6, "Supported Objects and Operations for DB2 for i"](#)
- [Section 1.7, "Non-Supported Objects and Operations for DB2 for i"](#)
- [Section 1.8, "Oracle GoldenGate Parameters Not Supported for DB2 for i"](#)
- [Section 1.9, "Supported Object Naming Conventions"](#)
- [Section 1.10, "Supported Character Sets"](#)

1.1 Overview of Oracle GoldenGate for IBM DB2 for i

With Oracle GoldenGate for IBM DB2 for i, you can replicate data to and from similar or dissimilar supported DB2 for i versions, or you can replicate 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 target DB2 for i database, Oracle GoldenGate can run directly on the DB2 for i target system or on a remote Windows or Linux system. If installed on a remote system, Replicat delivers the data by means of an ODBC connection, and no Oracle GoldenGate software is installed on the DB2 for i target.

Note: The DB2 for 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. For a list of other Oracle GoldenGate documentation, see ["Related Documents"](#)

1.2 Verifying Certification and System Requirements

Make sure that you are installing your product on a supported hardware or software configuration. For more information, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page.

Oracle has tested and verified the performance of your product on all certified systems and environments; whenever new certifications occur, they are added to the proper certification document right away. New certifications can occur at any time, and for this reason the certification documents are kept outside of the documentation libraries and are available on Oracle Technology Network.

1.3 Operating System Requirements

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

- [Section 1.3.1, "General Requirements"](#)
- [Section 1.3.2, "Memory Requirements"](#)
- [Section 1.3.3, "Disk Requirements"](#)
- [Section 1.3.4, "Network Requirements"](#)
- [Section 1.3.5, "Oracle GoldenGate Security Privileges"](#)

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.
- The IBM DB2 for i Program temporary fixes (PTFs) that are required by release for Oracle GoldenGate are detailed in the following tables:

IBM i7.1 Group PTF	Level	Name	Notes
SF99710	13037	Cumulative PTF	Other required PTF: 5770SS1, SI51060 Check with command: DSPPTF LICPGM(5770SS1) SELECT(SI51060)
SF99701	26	DB2 for i	
SF99709	99	Group HIPER	
SF99367	7	TCP/IP	

IBM i7.1 Group PTF	Level	Name	Notes
SF99572	12	JAVA	Java agent requires product 5761JV1 option 12 (Java SE 6 64-bit)

IBM i6.1 Group PTF	Level	Name	Notes
SF99610	13058	Cumulative PTF	Other required PTF: 5761SS1, SI51061 Check with command: DSPPTF LICPGM(5761SS1) SELECT(SI51061)
SF99601	30	DB2 for i	
SF99609	153	Group HIPER	
SF99354	15	TCP/IP	
SF99562	24	JAVA	Java agent requires product 5761JV1 option 12 (Java SE 6 64-bit)

IBM OS/400 V5R4 Group PTF	Level	Name	Notes
SF99540	12094	Cumulative PTF	Other required PTF: 5722SS1, SI50721, Check with command: DSPPTF LICPGM(5722SS1) SELECT(SI50721) If using the DB2 for i native command interface for OGG PASE_MAXSHR64 is set automatically so this message does not occur in that case
SF99504	33	DB2 for i	
SF99539	198	Group HIPER	
SF99315	22	TCP/IP	
SF99291	33	JAVA	Java agent requires product 5761JV1 option 10 (Java Developer Kit)

These required PTFs are the levels at which Oracle GoldenGate has been tested against for the 11g R2 and 12c releases. In order to check the group PTF levels, you must use the `WRKPTFGRP` command from a 5250 terminal session and check for the specific PTFs with the commands shown in the preceding tables. The specific extra PTFs must be at least temporarily applied.

1.3.2 Memory Requirements

The amount of memory that is required for Oracle GoldenGate depends on the amount of data being processed, the number of Oracle GoldenGate processes running, the amount of main storage (RAM, or physical memory) available to Oracle GoldenGate, and the amount of auxiliary storage (disk space, available as shared memory segments) that is available to Oracle GoldenGate for caching transaction data that exceeds available physical memory.

The amount of main storage 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.

On the DB2 for i platform, to provide enough shared memory segments to the Oracle GoldenGate cache manager, the recommended setting for the `PASE_MAXSHR64` environment variable is a value of 513 (128GB) or higher. If you use the DB2 for i native Oracle GoldenGate commands, `PASE_MAXSHR64` is set to provide 128GB of shared memory segments to the cache manager automatically. If not using the DB2 for i native commands, you can set this environment variable before starting the DB2 for i PASE session. For more information about evaluating Oracle GoldenGate memory requirements, see the `CACHEMGR` parameter in *Reference for Oracle GoldenGate for Windows and UNIX*.

Note: If `PASE_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 may ignore this message.

1.3.3 Disk Requirements

This section outlines the disk requirements for Oracle GoldenGate.

- To determine the size of the Oracle GoldenGate download file, view the Size column before downloading your selected build from Oracle Software Delivery Cloud. The value shown is the size of the files in compressed form. The size of the expanded Oracle GoldenGate installation directory will be significantly larger on disk. For more information, see [Section 2.2, "Understanding and Obtaining the Oracle GoldenGate Distribution"](#)
- Allow sufficient disk space for virtual memory. The default set by the Oracle GoldenGate cache manager is 64 GB on 64-bit systems. See [Section 1.3.2, "Memory Requirements"](#) for additional information about memory management.
- 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 *Administering Oracle GoldenGate for Windows and UNIX*.

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 *Administering Oracle GoldenGate for Windows and UNIX* 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.

1.3.5.1 Oracle GoldenGate Security Privileges on an DB2 for i Source 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. This person also must have authority to the RSTOBJ command, plus the ability to create a library if desired. For ease of installation, it is recommended that the user installing the product has *ALLOBJ authority

On an DB2 for i source system, the Manager and Extract processes are active. The DEFGEN utility also may be active if you are replicating data to a dissimilar target system. On an DB2 for i target system, the Replicat process is active unless you install Replicat on a remote Windows or Linux system. All processes run on both systems in a bidirectional configuration.

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 (DB2 for i *JOBCTL authority).

Assign *USE authority to all objects on the system that the Extract user profile must have access to. Assign *CHANGE authority to all objects on the system that the Replicat user profile must have access to. This can be accomplished by either granting *ALLOBJ authority to the user, or by setting the individual authority to the objects (FILE, LIBRARY and JOURNAL objects) that the user must access. This includes the objects in the QSYS2 library where the SQL catalog resides. These authorities must be granted through the native DB2 for i interface through a 5250 terminal session or through the DB2 for i Operations Navigator product available from IBM.

The Extract and Replicat database user profiles must be specified with the USERID parameter when you configure the parameter files (see [Chapter 4, "Configuring Oracle GoldenGate for DB2 for i"](#)) and in the DBLOGIN command prior to issuing any GCSCI commands that interact with the database.

1.3.5.2 Oracle GoldenGate Security Privileges on a Windows or Linux 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. On Windows, the person who installs Oracle GoldenGate must log in as Administrator.

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.

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]
```

The user profile must be specified with the `USERID` parameter when you configure the parameter files (see [Chapter 4, "Configuring Oracle GoldenGate for DB2 for i"](#)) and in the `DBLOGIN` command prior to issuing any GGSCI commands that interact with the database.

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.

Oracle GoldenGate supports timestamp data from 0001/01/03:00:00:00.000000 to 9999/12/31:23:59:59.999999. If a timestamp is converted from GMT to local time, these limits also apply to the resulting timestamp. Depending on the time zone, conversion may add or subtract hours, which can cause the timestamp to exceed the lower or upper supported limit.

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 DB2 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 with the maximum row length supported by the database.
- Tables that contain up to the maximum number of columns that is supported by the database, up to the maximum supported column size.
- DELETE FROM with no WHERE clause SQL statements 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.
- Both Library (Native) names including members, and SQL names are allowed.
- Partitioned tables
- Limitations on Automatic Heartbeat Table support are as follows:
 - The ADD HEARTBEATTABLE command creates a new file called ogghbfreq2 in the Oracle GoldenGate build folder. Do not delete this file because the pase heartbeat program reads the frequency values from it.
 - There is an extra executable in the Oracle GoldenGate build folder named ogghb.
 - An extra process named ogghb starts running from the time the ADD HEARTBEATTABLE command is given and runs until you disable the heartbeat with the DELETE HEARTBEATTABLE command. This process automatically restarts even if it is killed. To remove this process from the system, use the DELETE HEARTBEATTABLE command.
 - When using the ALTER HEARTBEATTABLE command to change the heartbeat frequency with the PURGE_FREQUENCY or RETENTION_TIME options, it takes approximately 60 + older 'frequency') seconds to be implemented.
 - There is an initial delay of 30 seconds between ADD HEARTBEATTABLE and the first record is updated in the heartbeat seed table.

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
- The Multi-Journal feature does not support multi-journal sync of a transaction across multiple journals.

1.8 Oracle GoldenGate Parameters Not Supported for DB2 for i

This section lists some of the Oracle GoldenGate configuration parameters that are not supported for the DB2 for i platform. For full descriptions of Oracle GoldenGate parameters and the databases they support, see Reference for Oracle GoldenGate for Windows and UNIX.

BATCHSQL (not supported on V5R4 and i6.1 only)
 BR
 ASCIITOEBCDIC and EBCDICTOASCII

BINARYCHARS
LOBMEMORY
TRAILCHARSETEBDIC
Any encryption options that use AES encryption

1.9 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 DB2 for 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 DB2 for i system. For SQL names, Oracle GoldenGate supports the wildcarding of table names, but not schema names. For instructions on wildcarding SQL names, see *Administering Oracle GoldenGate for Windows and UNIX*.

1.10 Supported Character Sets

In all prior releases of the DB2 for i, Extract all text data was converted to Unicode. Either UTF-8 for single and multi-byte CCSIDs or UTF-16 for double byte CCSIDs. For SQL, that means all non-binary CHAR, VARCHAR and CLOB data would be converted to UTF-8 and all GRAPHIC, VARGRAPHIC and DBCLOB data would be converted to UTF-16 for the trail, either for initial loads or "normal" extracts. This is still true if using a trail format in Extract that is lower than 12.2.

The behavior of `defgen` prior to version 12.2 for versions that had a column charset column, was to indicate -1 which represented that the column characters set was the default for columns that used character sets. The trail header would then be used to indicate the character set for the column data which was always set to UTF-8/UTF16.

Starting with Oracle GoldenGate Version 12.2 the default behavior has changed to allow character sets that are supported by Oracle GoldenGate conversions to pass through unchanged. This change automatically will reduce the CPU consumption and increase the throughput rate of extracts proportional to the amount of text data in the records being processed. If a CCSID is found in the extract that Oracle GoldenGate cannot convert, the extract will default to its original behavior for that column and convert the text data to the appropriate Unicode charset as it did in prior releases. This ensures that any replicat that processes a trail from the new extract is capable of handling the text data from the extract.

If the original behavior of the IBM i extract is desired then the keyword `TRAILCHARSETUNICODE` must be added to the extract `prm` file. This will cause all text data to be converted to unicode as it was in the prior releases of Oracle GoldenGate for DB2 for i. Alternatively it is possible to selectively revert to the old conversion behavior at the specific object or even column level by using the keyword `COLCHARSET` on the table definition.

Examples in a typical `prm` file for a "normal" (non-initial load) extract:

```
TRAILCHARSETUNICODE
```

- all text data in objects included in all table statements will be converted to Unicode
- if this keyword is not included the extract will not convert text data by default

.

```
table <schema>.<table>, COLCHARSET(ALL, UTF-8)
```

or

```
table <schema>.<table>, charset(UTF-8)
```

- all text data in objects that match this specific table statement will be converted to Unicode (double byte columns will be UTF-16)

.

```
table <schema>.<table>, COLCHARSET(TXTCOL4, UTF-8)
```

- only `TXTCOL4` will be converted to UTF-8 all other text data will pass through unchanged

.

Note: Due to how the IBM for i PASE database layer functions as well as Oracle GoldenGate's internal processing, there are certain situations where Unicode conversion is still required.

.

- Initial Load extracts will automatically convert all data to Unicode and indicate Unicode data in the columns]

- Extracts that use a trail format that is prior to the Oracle GoldenGate 12.2, the extracts will automatically fall back to converting text data to Unicode

- table specifications that include any column functions, or `SQLEXEC`, `FETCHCOLUMNS` or `FETCHMODCOLS` will require that either `TRAILCHARSETUNICODE` is specified or the specific tables or columns are changed to include the `COLCHARSET` modifier. This is true if replicat is using column functions as well. In that case the extract table(s) that map to the replicat tables must be sent as Unicode.

.

For defgen `TRAILCHARSETUNICODE` is not supported since defgen does not generate a trail, but if you are using `TRAILCHARSETUNICODE` or overriding the column charset(s) and are using a defs file on replicat then defgen must also specify the equivalent column charset overrides:

.

```
table <schema>.<table>, COLCHARSET(ALL, UTF-8)
```

or

table <schema>.<table>, charset(UTF-8)

- all text data in objects that match this specific table statement will be converted to Unicode (double byte columns will be UTF-16)

.

table <schema>.<table>, COLCHARSET(TXTCOL4, UTF-8)

- only TXTCOL4 will be converted to UTF-8 all other text data will pass through unchanged

Also note that in Oracle GoldenGate version 12.2, the default behavior is to have column metadata included in the trail data inline with the operation data as required. This means that definitions files are no longer needed by Oracle GoldenGate replication and will be ignored.

If however the Oracle GoldenGate Replicats or pumps indicate that they choose to override this behavior and use the definitions files, or a prior trail format level is used, the `defgen` definitions files will need to be recreated. In this case, care must be taken to ensure that the definitions match what extract is writing to the trails. Therefore any `TRAILCHARSETUNICODE` keywords, `CHARSET` or `COLCHARSET` modifiers that exist in the extract `prm` file must also exist in the `defgen prm` file that matches the extract.

`ASSUMETARGETDEFS OVERRIDE` in Replicat also requires that the extract use `TRAILCHARSETUNICODE` or the equivalent `COLCHARSET` modifiers on the tables in the extract parameter file. This is due to replicat still connecting to the database with a Unicode connection and therefore internally currently treats all text fields as Unicode.

Installing Oracle GoldenGate

This chapter contains instructions for installing Oracle GoldenGate on a DB2 for i source system and on a remote Windows or Linux target system for delivery to a DB2 for i target.

This chapter contains the following sections:

- [Section 2.1, "Installation Overview"](#)
- [Section 2.2, "Understanding and Obtaining the Oracle GoldenGate Distribution"](#)
- [Section 2.3, "Setting Library Paths for Dynamic Builds on DB2 for i"](#)
- [Section 2.4, "Installing Oracle GoldenGate on DB2 for i"](#)
- [Section 2.5, "Installing Oracle GoldenGate on Linux"](#)
- [Section 2.6, "Installing Oracle GoldenGate on Windows"](#)

These instructions are for installing Oracle GoldenGate for the first time. Additionally, they are for downloading the base release of a new version of Oracle GoldenGate.

To download and install subsequent patches to the base release, go to the Patches and Updates tab of My Oracle Support at:

<http://support.oracle.com>

To upgrade Oracle GoldenGate from one version to another, follow the upgrade instructions in *Upgrading Oracle GoldenGate for Windows and UNIX*.

2.1 Installation Overview

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.

Oracle GoldenGate for DB2 for i is installed directly on a source DB2 for i system. Oracle GoldenGate for DB2 for i can be installed either directly on an DB2 for i target system or on a Windows or Linux system for remote delivery to the DB2 for i system. When installed for remote delivery, Replicat connects from the Windows or Linux system to the DB2 for i system through ODBC. A database on the Windows or Linux system is not required.

2.2 Understanding and Obtaining the Oracle GoldenGate Distribution

For complete information about how to obtain Oracle Fusion Middleware software, see "Understanding and Obtaining Product Distributions" in *Planning an Installation of Oracle Fusion Middleware*.

To download the Oracle WebLogic Server and Coherence software for development or evaluation, see the following location on the Oracle Technology Network (OTN):

<http://www.oracle.com/technetwork/middleware/fusion-middleware/downloads/index.html>

For more information about locating and downloading Oracle Fusion Middleware products, see the *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files* on OTN.

To obtain Oracle GoldenGate follow these steps:

1. Go to Oracle Technology Network.
2. Find the Oracle GoldenGate 12c (12.2.0.1) release.
3. Download the ZIP file appropriate for your system, a DB2 for i target system or a DB2 for i remote delivery to the system.

2.3 Setting Library Paths for Dynamic Builds on DB2 for i

Oracle GoldenGate uses shared libraries. When you install Oracle GoldenGate on an DB2 for 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 in PASE through QP2TERM or ssh on an DB2 for 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?
\$ users/ogg > ./ggsci	No
\$ users > ./ogg/ggsci	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
```

To Set the Variables from a 5250 Terminal Command Prompt

Set these variables prior to running QP2TERM

```
ADDENVVAR ENVVAR(PATH) VALUE('installation_directory:$PATH')
ADDENVVAR ENVVAR(PASE_LIBPATH) VALUE('installation_directory')
```

Note: When running the DB2 for i native commands, all environment setup is performed automatically by the command-based on the Oracle GoldenGate installation library used to run the command from.

Note: To view the libraries that are required by an Oracle GoldenGate process, use the `dump -Hv -X64` process shell command before starting the process.

2.4 Installing Oracle GoldenGate on DB2 for i

Follow these steps to install Oracle GoldenGate on an DB2 for i system.

Note: The user profile running the install must have authority to the RSTOBJ command.

1. On the system where Oracle GoldenGate is to be installed, create a directory for Oracle GoldenGate.

```
- MKDIR DIR('/GoldenGate')
```

2. You can create a library for Oracle GoldenGate on the installation system, or you can create it through the installation script that you will run later in these steps.

```
- CRTLIB LIB(goldengate) TEXT('Oracle GoldenGate Product Library') ASP(1)
```

3. Unzip the downloaded file on a Windows or Linux or UNIX system.
4. FTP the resulting tar file from that system to the folder that you created on the DB2 for i installation system.

```
ftp IBMi_IP_address
.
User (system:(none)):userid
.
331 Enter password.
.
Password: password
.
230 userid logged on.
.
ftp> bin
.
ftp> cd goldengate
.
ftp> put install_file
.
ftp> quit
```

5. (If you created a library) From a 5250 terminal session, change your current library to the Oracle GoldenGate library.

```
CHGCURLIB Oracle_GoldenGate_ library
```

6. Run a QP2TERM terminal session.

```
- CALL QP2TERM
```

7. Extract the installation objects from the tar file.

```
tar -xf tar_file
```

8. In the Oracle GoldenGate directory, run the shell script ggos400install.

```
ggos400install goldengate
```

The default is to install the required objects into the current library (set in the preceding steps), but you can create a library by using the `-c` option. Additional options are available.

Note: There must be a separate Oracle GoldenGate library for each Oracle GoldenGate directory. The install script checks for this condition and will prevent installation to the same library that another installation is using. The reason for this is to prevent mismatches between the Oracle GoldenGate installation and the OGGPRCJRN *SRVPGM object.

Syntax:

```
./ggos400install [-h] [-f] [-u userid] [[-a aspname] | [-n aspnun]] [-c|-l  
library name]
```

Options:

- `-h` shows this usage help.
- `-f` forces a change to a new installation library. This argument only affects an existing installation.
- `-u userid` specifies the userid that will own the installation.
- `-a aspname` specifies the name of the ASP where objects will be restored. If no `aspname` is provided, the system asp is assumed. This option cannot be used with `-n`.
- `-n aspnun` specifies the number of the user asp where the objects will be restored. This option cannot be used with `-a`.
- `-c library` specifies the name of the library where the objects will be restored. The library will be created.
- `-l library` specifies the name of the library where the objects will be restored. The library must exist. If a library is not specified for a new installation, the installer will attempt to use the current library of the user that is running the installer. If a library is not specified for an existing installation, the installer will attempt to use the library that is set in the `oggprcjrn.srvpgm` link.

Note: If Oracle GoldenGate is reinstalled, you must run `ggos400install` again. On a reinstall, `ggos400install` will recognize the prior configuration, so no arguments are needed. If the `oggprcjrn.srvpgm` link is changed or removed, `ggos400install` must be run again with the Oracle GoldenGate installation library specified by the link.

9. Exit QP2TERM.

- F3

Note: On an DB2 for i system, it is not necessary to create any working directories in the Oracle GoldenGate installation directory. The `ggos400install` script performs this task.

2.5 Installing Oracle GoldenGate on Linux

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

1. Extract the Oracle GoldenGate download 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

2.6 Installing Oracle GoldenGate on Windows

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

2.6.1 Installing Oracle GoldenGate into a Windows Cluster

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

2.6.2 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
```
6. Copy the following files from the Oracle GoldenGate installation directory to the SYSTEM32 directory.

```
category.dll  
ggsmg.dll
```

2.6.3 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.

2.6.4 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.
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. ¹ 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.
PASSWORD <i>password</i>	Specifies the password for the user that is specified with <code>USER</code> .

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

4. 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.

Preparing the System for Oracle GoldenGate

This chapter contains guidelines for preparing the DB2 for i system to support Oracle GoldenGate.

This chapter contains the following sections:

- [Section 3.1, "Preparing the Journals for Data Capture by Extract"](#)
- [Section 3.2, "Specifying Object Names"](#)
- [Section 3.3, "Preparing Tables for Processing"](#)
- [Section 3.4, "Adjusting the System Clock"](#)
- [Section 3.5, "Configuring the ODBC Driver"](#)

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 *Administering Oracle GoldenGate for Windows and UNIX*.

3.1.2 Setting Journal Parameters

To support the capture of data by the Extract process, the following are the minimal journaling parameter settings that 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

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.

If using another application that is using the journals that Oracle GoldenGate will be reading, consideration must be given regarding any automatic journal receiver cleanup that may be in place. Oracle GoldenGate must be able to read the journal receivers before they are detached or removed.

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
```

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 DB2 for 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. For a SQL name only the first member is used.

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.

When specifying a native table name in a MAP statement on a platform other than DB2 for i, the name must be enclosed within double quotes so that Oracle GoldenGate correctly interprets it as a separator character.

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 the native system 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. Unique access path contained in the physical file object.
4. 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 *Reference for Oracle GoldenGate for Windows and UNIX*.

Note: If you want to use the RRN of the records as the key for a table, you may access the GGHEADER Oracle GoldenGate Environment Variable AUDITRBA which will contain the RRN for each record processed.

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.

```
STOP EXTRACT group
```

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

```
INFO EXTRACT group
```

3. Make the change to the key.

4. Start Extract.

```
START EXTRACT group
```

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, USERID specifies the Extract user profile, and PASSWORD specifies that profile's password. See *Reference for Oracle GoldenGate for Windows and UNIX* 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 `oggprcjrn` 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/oggprcjrn.srvpgm oggprcjrn.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 DB2 for 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

This section applies only if you installed Replicat on a Windows or Linux system to operate remotely from the DB2 for i target. In this configuration, Replicat must 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:

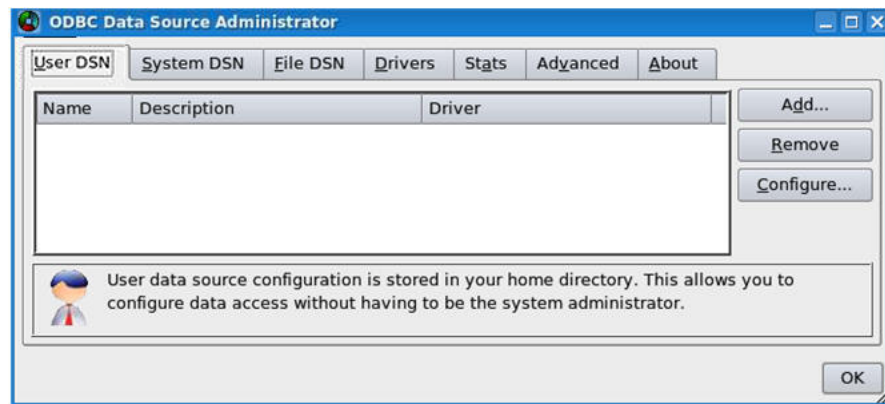
```
rpm -ivh iSeriesAccess-7.1.0-1.0.i386.rpm
```

64-bit driver:

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

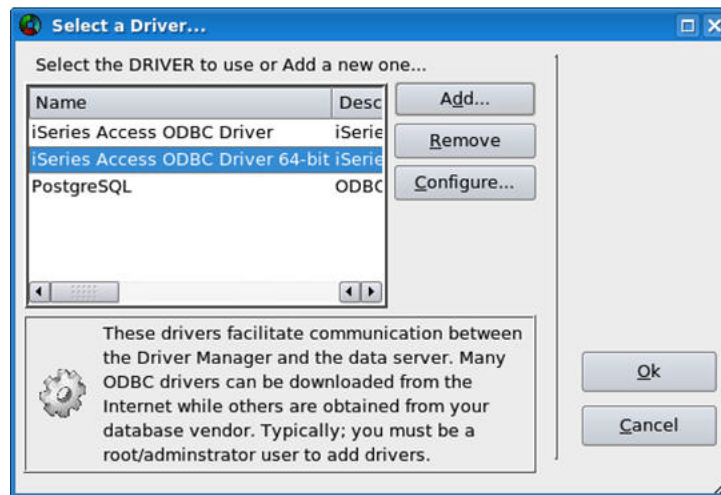
3. You can create a user DSN (a connection that is available only to the user that created it) or a system DSN (a connection that is available to all users on the system). To create a user DSN, log on to the system as the user that you will be using for the Replicat process.
4. Run the ODBC configuration utility.
5. 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

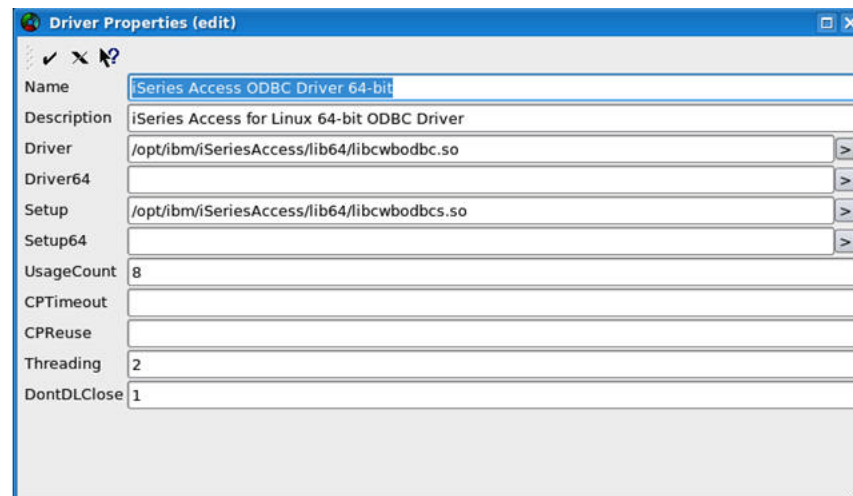


This figure shows the User DSN tab of the ODBC Data Source Administrator dialog.

6. On the tab you selected, click **Add**.
7. 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

This figure shows the Select a Driver... dialog.

Figure 3–3 Manually Editing Driver Properties When the Driver is Not Found

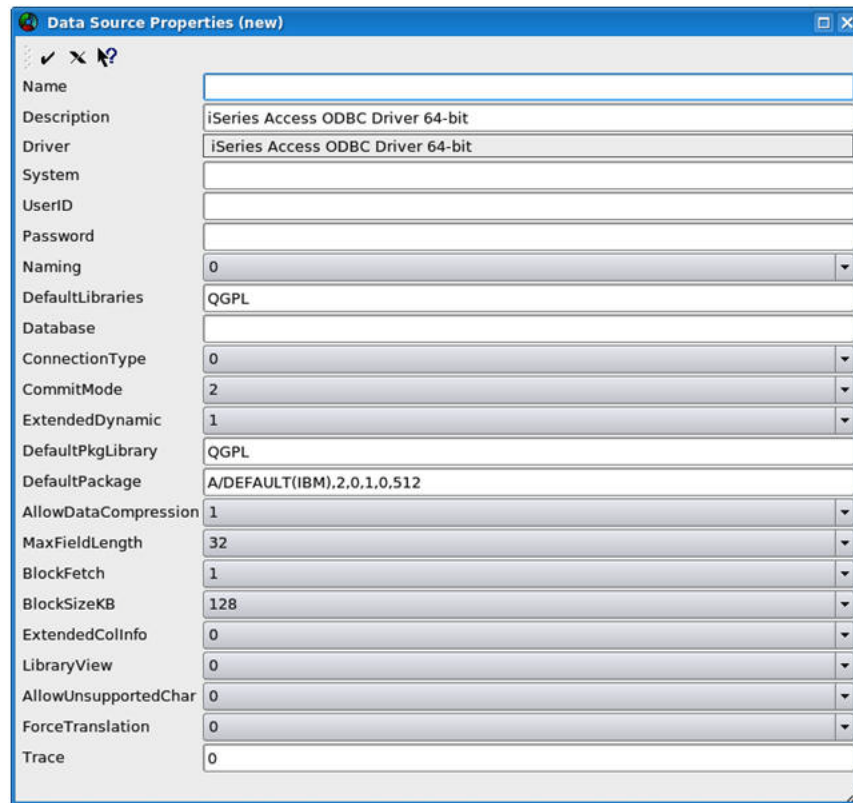
This figure shows the fields on the Driver Properties (edit) dialog.

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 DB2 for 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



This figure shows the fields on the Data Source Properties dialog

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 DBCLOB 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. See *Reference for Oracle GoldenGate for Windows and UNIX* for detailed syntax.

```
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.

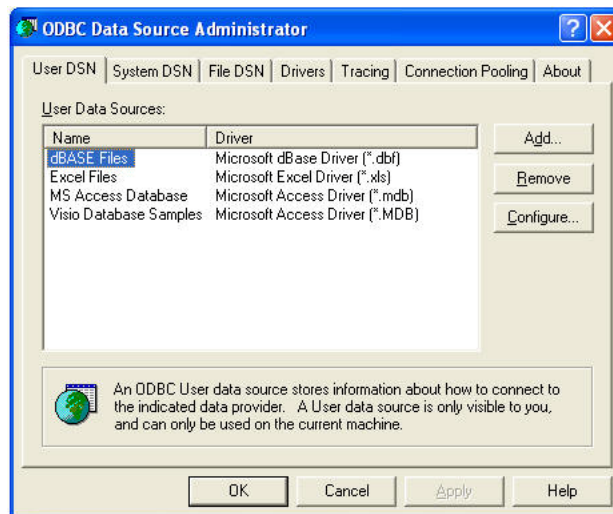
Note: Only BLOWFISH encryption is supported for DB2 for i systems.

3.5.2 Configuring ODBC on Windows

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

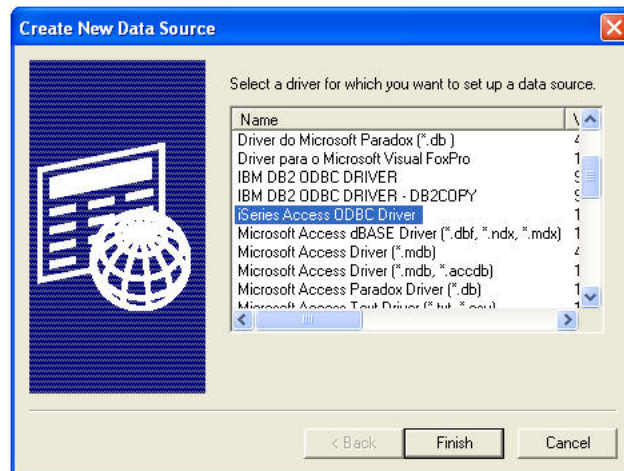
1. Download and install the 32-bit or 64-bit iSeries Access ODBC driver from the DB2 for iSeries Access package on the remote Windows system according to the vendor documentation. The iSeries ODBC driver is supplied as a free component of iSeries Access.
2. You can create a user DSN (a connection that 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



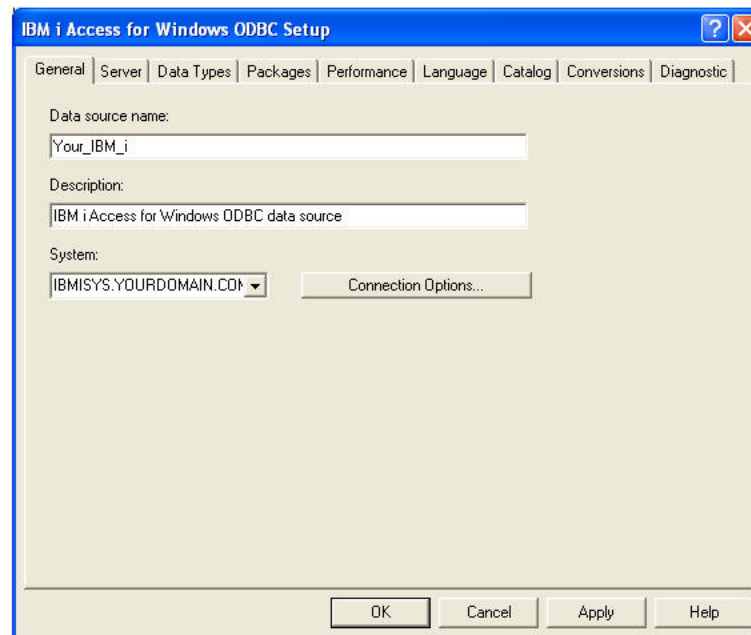
This figure shows the ODBC Data Source Administrator dialog

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

This figure shows the iSeries Access ODBC driver from the list of drivers

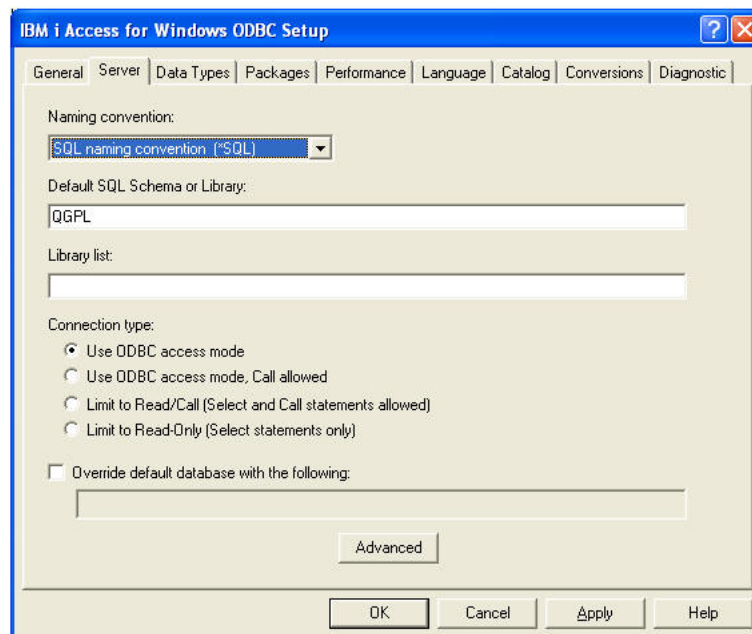
- *****
7. On the General tab of the DB2 for i Access for Windows ODBC Setup dialog, provide a name (without any spaces) in the **Data Source Name** field, add an optional description in the **Description** field, and then select the system name from the **System** selection list.

Figure 3–7 Setting General ODBC Properties on Windows

This figure shows the General tab of the DB2 for i Access for Windows ODBC setup dialog

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

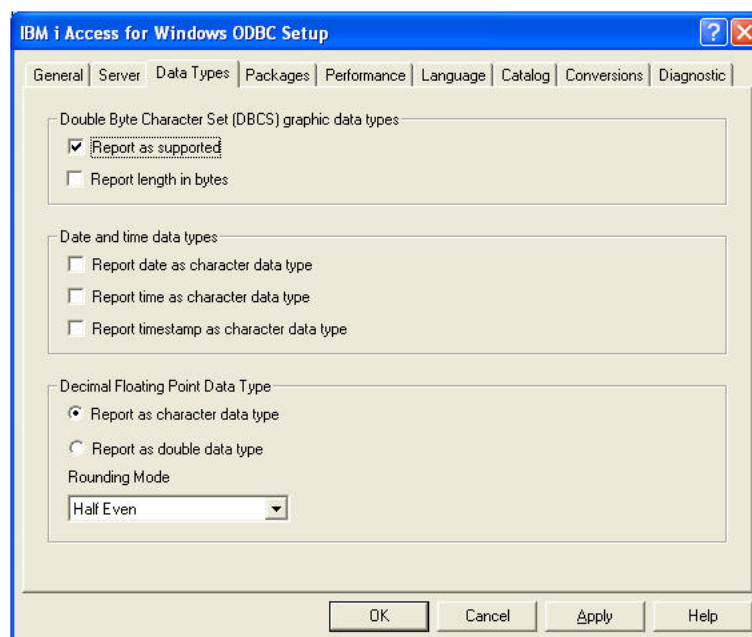
Figure 3–8 Setting Server ODBC Properties on Windows



This figure shows the Server tab of the IBM i Access for Windows ODBC Setup

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

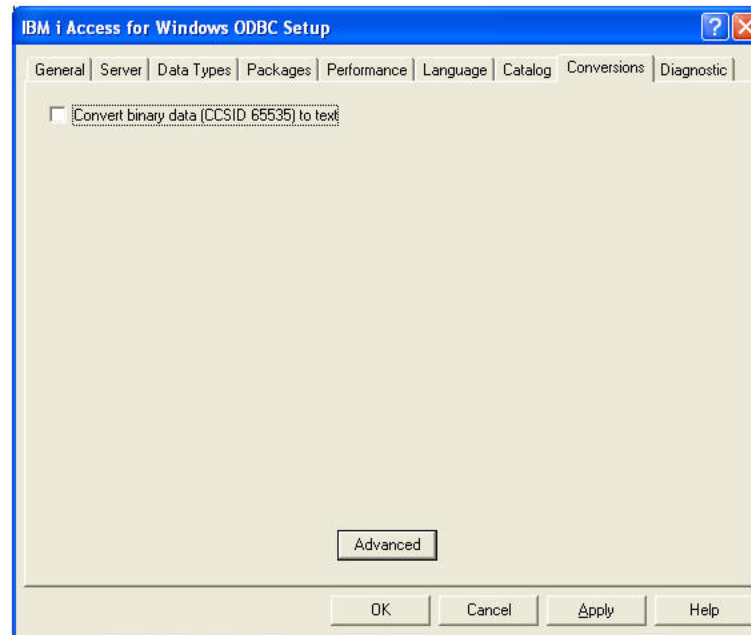
Figure 3–9 Setting DBCS ODBC Properties



This figure shows the Data Type tab, select the Report as supported checkbox under the DBCS graphic data types

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

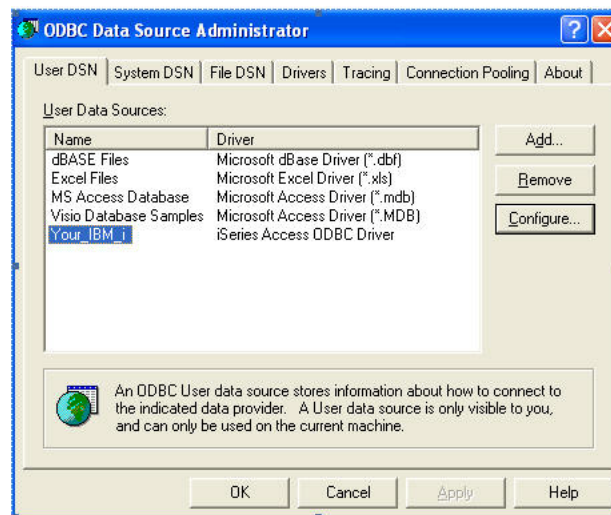
Figure 3–10 Setting Conversion ODBC Properties



This figure shows the Conversions tab to clear the Convert binary data (CCSID 65535) to text check box

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



This figure shows the ODBC Data Source Administrator Dialog. Confirm the new Data Source Name appears under User Data Sources. Click OK to exit the ODBC config utility.

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. See *Reference for Oracle GoldenGate for Windows and UNIX* for detailed syntax.

```
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.

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

Configuring Oracle GoldenGate for DB2 for i

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

This chapter contains the following sections:

- [Section 4.1, "What to Expect from this Procedure"](#)
- [Section 4.2, "Getting Started with Oracle GoldenGate"](#)
- [Section 4.3, "Creating the Oracle GoldenGate Instance"](#)
- [Section 4.4, "Creating a GLOBALS File"](#)
- [Section 4.5, "Creating a Data Definitions File"](#)
- [Section 4.6, "Encrypting the Extract and Replicat Passwords"](#)
- [Section 4.7, "Configuring Extract for Change Capture from DB2 for i"](#)
- [Section 4.8, "Configuring Replicat for Change Delivery to DB2 for i"](#)
- [Section 4.9, "Next Steps in the Deployment"](#)
- [Section 4.10, "When to Start Replicating Transactional Changes"](#)
- [Section 4.11, "Testing Your Configuration"](#)

4.1 What to Expect from this Procedure

These instructions show you how to configure a set of basic 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 to 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.

This chapter is a subset of the instructions in *Administering Oracle GoldenGate for Windows and UNIX*. It focuses on the basic parameters that are specific to DB2 for i.

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 Getting Started with Oracle GoldenGate

Before proceeding with the configuration process, you should get familiar with the Oracle GoldenGate architecture, the command interface, and the methods for supplying input and instructions to the processes. See *Administering Oracle GoldenGate for Windows and UNIX* for this information.

4.3 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. For details on configuring Manager and its network connections, see *Administering Oracle GoldenGate for Windows and UNIX*.

4.4 Creating a GLOBALS File

The GLOBALS parameter file contains parameters that affect all processes within an Oracle GoldenGate instance. The GLOBALS parameter NAMECCSID is specific to DB2 for i and may be required if the SQL catalog contains object names that are referenced by a different CCSID than the system CCSID. The SQL catalog is created in the system CCSID and does not indicate this difference when queried. Oracle GoldenGate makes queries to the catalog and could retrieve the name incorrectly unless NAMECCSID is used to supply the correct CCSID value. For more information, see *Reference for Oracle GoldenGate for Windows and UNIX*.

4.5 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 *Administering Oracle GoldenGate for Windows and UNIX*. This file must be in place on the target system before you start the Oracle GoldenGate processes for the first time.

4.6 Encrypting the Extract and Replicat Passwords

It is strongly recommended that you encrypt the passwords of the user profiles that will be used for the primary and data pump Extracts, and for the Replicat process. Oracle GoldenGate must use Blowfish encryption on the DB2 for i platform. The standard Oracle GoldenGate encryption method of AES (Advanced Encryption Standard) is not supported by the IMB i platform. To encrypt the password, see *Administering Oracle GoldenGate for Windows and UNIX*. That documentation also contains information about how to encrypt data within disk storage and across TCP/IP.

Note: The Oracle GoldenGate credential store is not supported by the iSeries platform.

4.7 Configuring Extract for Change Capture from DB2 for i

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

4.7.1 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.

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
SOURCEDB FINANCE USERID ogg, PASSWORD AACAAAAAAAAAAAA, BLOWFISH ENCRYPTKEY mykey
ENCRYPTTRAIL
EXTTRAIL /ggs/dirdat/lt
TABLE hr.*;
```

Table 4–1 Basic Parameters for the Primary Extract

Parameter	Description
EXTRACT <i>group</i>	<i>group</i> is the name of the Extract group.
SOURCEDB <i>database id</i> , PASSWORD <i>password</i> , BLOWFISH ENCRYPTKEY <i>keyname</i>	<p>Specifies database connection information.</p> <ul style="list-style-type: none"> ■ 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. ■ USERID specifies the Extract database user profile. ■ PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command. Enter or paste the encrypted password after the PASSWORD keyword. ■ BLOWFISH ENCRYPTKEY <i>keyname</i> specifies the name of the lookup key in the local ENCKEYS file. <p>See <i>Administering Oracle GoldenGate for Windows and UNIX</i> for encryption options.</p>
ENCRYPTTRAIL	Encrypts the local trail with Blowfish encryption. For more information, see the Oracle GoldenGate security documentation in <i>Administering Oracle GoldenGate for Windows and UNIX</i> .
EXTTRAIL <i>pathname</i>	Specifies the path name of the local trail to which the primary Extract writes captured data.
TABLE <i>owner.table</i> ;	<p>Specifies a table or tables for which to extract data changes. For supported object names, see <i>Administering Oracle GoldenGate for Windows and UNIX</i>.</p> <p>Terminate the TABLE statement with a semi-colon.</p> <p>To exclude tables from a wildcard specification, use the TABLEEXCLUDE parameter after the TABLE statement.</p> <p>For more information about parameters and options that control data filtering, mapping, and manipulation, see <i>Reference for Oracle GoldenGate for Windows and UNIX</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.

4.7.2 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.

1. 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.

2. 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 AACAAAAA, 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</i>	<i>group name</i> is the name of the data pump.
SOURCEDB <i>database</i> USERID <i>user</i> , PASSWORD <i>password</i> , BLOWFISH ENCRYPTKEY <i>keyname</i>	<p>Specifies database connection information.</p> <ul style="list-style-type: none"> ■ SOURCEDB specifies the <i>default</i> DB 2 for i database. ■ USERID specifies the Extract database user profile. ■ PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command. Enter or paste the encrypted password after the PASSWORD keyword. ■ BLOWFISH ENCRYPTKEY <i>keyname</i> specifies the name of the lookup key in the local ENCKEYS file. <p>See <i>Administering Oracle GoldenGate for Windows and UNIX</i> for encryption options.</p>
DECRYPTTRAIL BLOWFISH	Decrypts the input trail.
RMTHOST <i>hostname</i> , MGRPORT <i>portnumber</i>	<ul style="list-style-type: none"> ■ RMTHOST specifies the name or IP address of the target system. ■ MGRPORT specifies the port number where Manager is running on the target. <p>For additional options and encryption details, see <i>Administering Oracle GoldenGate for Windows and UNIX</i>.</p>
ENCRYPTTRAIL BLOWFISH	Encrypts the remote trail with Blowfish encryption. For more information, see the Oracle GoldenGate security documentation in <i>Administering Oracle GoldenGate for Windows and UNIX</i> .
RMTTRAIL <i>pathname</i>	Specifies the path name of the remote trail.
TABLE <i>owner.table</i> ;	<p>Specifies a table or tables to process. For supported object names, see <i>Administering Oracle GoldenGate for Windows and UNIX</i>.</p> <p>Terminate the TABLE statement with a semi-colon.</p> <p>To exclude tables from a wildcard specification, use the TABLEEXCLUDE <i>owner.table</i> parameter after the TABLE statement.</p> <p>For more information and for additional options that control data filtering, mapping, and manipulation, see TABLE in <i>Reference for Oracle GoldenGate for Windows and UNIX</i>.</p>

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

4.8 Configuring Replicat for Change Delivery to DB2 for i

These steps configure Replicat to apply data to a DB2 for i target database, operating either on the target system or on a remote Windows or Linux system. To configure Replicat for change delivery to a different database type, such as an Oracle database, follow the directions in the Oracle GoldenGate Installation and Configuration guide for that database. There may be additional parameters and requirements for delivery to that database type.

Note: There does not have to be a database on a Windows or Linux machine to support connection by ODBC by Replicat.

4.8.1 Creating a Checkpoint Table

Replicat maintains its checkpoints in a checkpoint table in the DB2 for i target database. Each checkpoint is written to the checkpoint table, that must be journaled, 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.

A common method of create the checkpoint table with journaling is as follows:

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

```
DEFAULTJOURNAL library_name/journal_name
```

Where: *library_name* is the name of the library and *journal_name* is the name of the default journal.

2. Add the checkpoint table.

```
ADD CHECKPOINTTABLE library_name.chkptab
```

```
Successfully created checkpoint table kgr.chkptab
```

3. Add journaling to the checkpoint table.

```
ADD TRANDATA library_name.CHKPTAB
```

For more information about creating a checkpoint table, see *Administering Oracle GoldenGate for Windows and UNIX*.

4.8.2 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 *Reference for Oracle GoldenGate for Windows and UNIX*.

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

```
EDIT PARAMS name
```

Where: *name* is the name of the Replicat group.

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

```
REPLICAT financer
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</i>	<i>group</i> is the name of the Replicat group.
TARGETDB <i>database</i> USERID <i>user</i> , PASSWORD <i>password</i> , BLOWFISH ENCRYPTKEY <i>keyname</i>	<p>Specifies database connection information.</p> <ul style="list-style-type: none"> ■ SOURCEDB specifies the data source name (DSN) of the target DB2 for i database. ■ USERID specifies the Replicat database user profile. ■ PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command. Enter or paste the encrypted password after the PASSWORD keyword. ■ BLOWFISH ENCRYPTKEY <i>keyname</i> specifies the name of the lookup key in the local ENCKEYS file.
DECRYPTTRAIL BLOWFISH	Decrypts the input trail.
SOURCEDEFS <i>pathname</i> ASSUMETARGETDEFS	<p>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 4-2. Use ASSUMETARGETDEFS if the source and target tables are all DB2 for i and have the same definitions.</p>
MAP <i>owner.table</i> , TARGET <i>owner.table</i> ;	<p>Specifies a relationship between a source and target table or tables. The MAP clause specifies the source objects, and the TARGET clause specifies the target objects to which the source objects are mapped.</p> <ul style="list-style-type: none"> ■ <i>owner</i> is the schema or library name. ■ <i>table</i> is the name of a table or a wildcard definition for multiple tables. <p>For supported object names, see <i>Administering Oracle GoldenGate for Windows and UNIX</i>.</p> <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 Reference for Oracle GoldenGate for Windows and UNIX.</p>

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

4.9 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 *Administering Oracle GoldenGate for Windows and UNIX*. 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 *Reference for Oracle GoldenGate for Windows and UNIX*.

4.10 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 capture and apply processes to handle ongoing transactional changes while an initial load is being applied to the target. This process is known as *initial synchronization*, or also as *instantiation*. 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.

See *Administering Oracle GoldenGate for Windows and UNIX* for instantiation options.

4.10.1 Starting Extract During Instantiation

When Extract starts for the first time to begin capturing data during the instantiation process, 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.

4.10.2 Changing the Position of Extract to a Later Time

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 *Reference for Oracle GoldenGate for Windows and UNIX*.

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.

4.11 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.

This chapter contains the following sections:

- [Section 5.1, "About the Instantiation Process"](#)
- [Section 5.2, "Overview of Basic Oracle GoldenGate Instantiation Steps"](#)
- [Section 5.3, "Satisfying Prerequisites for Instantiation"](#)
- [Section 5.4, "Making the Instantiation Procedure More Efficient"](#)
- [Section 5.5, "Configuring the Initial Load"](#)
- [Section 5.6, "Adding Change-Capture and Change-Delivery processes"](#)
- [Section 5.7, "Performing the Target Instantiation"](#)
- [Section 5.8, "Monitoring Processing after the Instantiation"](#)
- [Section 5.9, "Backing up Your Oracle GoldenGate Environment"](#)
- [Section 5.10, "Positioning Extract After Startup"](#)

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 Your 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 for DB2 for i](#)" 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 *Administering Oracle GoldenGate for Windows and UNIX*.

5.5 Configuring the Initial Load

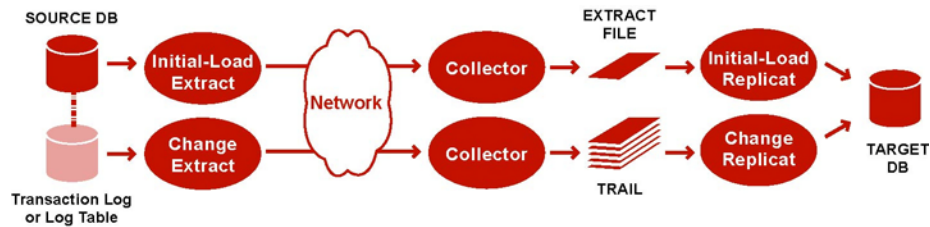
Oracle GoldenGate supports the following load methods specifically for Oracle:

- [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

Figure 5–1 Configuring an Initial Load from File to Replicat



This graphic shows loading data 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 DB2 for 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"> ■ SOURCEDB specifies the name of the source database. ■ USERID specifies the Extract database user profile. ■ PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command (see "Encrypting the Extract and Replicat Passwords" on page 4-3). Enter or paste the encrypted password after the PASSWORD keyword. ■ BLOWFISH ENCRYPTKEY <i>keyname</i> specifies the name of the lookup key in the local ENCKEYS file.
RMTHOST <i>hostname</i> , MGRPORT <i>portnumber</i> , [<i>encryption options</i>]	<ul style="list-style-type: none"> ■ RMTHOST specifies the name or IP address of the target system. ■ MGRPORT specifies the port number where Manager is running on the target. ■ <i>encryption options</i> specifies optional encryption of data across TCP/IP. <p>For additional options and encryption details, see <i>Reference for Oracle GoldenGate for Windows and UNIX</i>.</p>
ENCRYPTTRAIL BLOWFISH KEYNAME <i>keyname</i>	Encrypts the remote file with Blowfish encryption. For more information about security, see <i>Administering Oracle GoldenGate for Windows and UNIX</i> .
RMTFILE <i>path name</i> , [MEGABYTES <i>n</i>]	<p>Specifies the remote file to which the load data will be written. Oracle GoldenGate creates this file during the load.</p> <p>Note: The size of an extract file cannot exceed 2GB.</p> <ul style="list-style-type: none"> ■ <i>path name</i> is the relative or fully qualified name of the file. ■ MEGABYTES designates the size of each file.
TABLE <i>owner.table</i> ;	<p>Specifies a source table or tables for initial data extraction.</p> <ul style="list-style-type: none"> ■ <i>owner</i> is the library or schema name. ■ <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.

4. Enter any appropriate optional Extract parameters listed in *Reference for Oracle GoldenGate for Windows and UNIX*.

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.

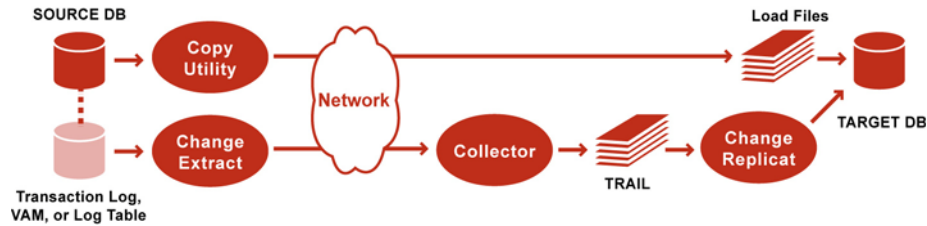
Table 5–2 Initial Load Replicat Parameters for Loading Data from File to Replicat

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> , algorithm ENCRYPTKEY <i>keyname</i>	Specifies database connection information. <ul style="list-style-type: none"> 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. USERID specifies the Replicat database user profile. PASSWORD specifies the user's password that was encrypted with the ENCRYPT PASSWORD command (see "Encrypting the Extract and Replicat Passwords" on page 4-3). Enter or paste the encrypted password after the PASSWORD keyword. algorithm ENCRYPTKEY <i>keyname</i> specifies the encryption method and keyname that was specified in the ENCRYPT PASSWORD command.
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> <ul style="list-style-type: none"> <i>path name</i> is the relative or fully qualified name of the file or trail. Use EXTTRAIL only if you used the MAXFILES option of the RMTFILE parameter in the Extract parameter file. 	Specifies the input extract file specified with the Extract parameter RMTFILE.
SOURCEDEFS <i>file name</i> ASSUMETARGETDEFS <ul style="list-style-type: none"> 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. Use ASSUMETARGETDEFS if the source and target tables have the same definitions. 	Specifies how to interpret data definitions.
MAP <i>owner.table</i> , TARGET <i>owner.table</i> ; <ul style="list-style-type: none"> <i>owner</i> is the schema name. <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. 	Specifies a relationship between a source and target table or tables.

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

5.5.2 Configuring an initial load with a database utility

Figure 5–2



This graphic shows loading from file to Replicat.

This graphic shows the parallel flows of the initial load and the ongoing capture and replication of transactional changes during the load period. The copy utility writes the data to a file, which is loaded to the target. Meanwhile, an Extract process captures change data and sends it to a trail on the target for Replicat to read and apply to the target.

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 DB2 for 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 DB2 for 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 *specific* 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\dir\dat\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.
3. On the source system, start the primary and data pump Extract groups to start change extraction.

```
START MANAGER

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: For information about monitoring processes, see *Administering Oracle GoldenGate for Windows and UNIX*.
- 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 Your 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 *Reference for Oracle GoldenGate for Windows and UNIX*.

Note: Because the extract will be synchronizing all of the journals in the extract by system sequence number because it is possible for a transaction to be split across them, if a given journal is independently repositioned far into the past, the resulting latency from reprocessing the entries will cause the already-read journals to stall until the reading of the latent journal catches up.

Using Remote Journal

This chapter contains instructions for remote journal preparation and adding a remote journal. Remote Journal support in the IBM DB2 for i operating system provides the ability for a system to replicate, in its entirety, a sequence of journal entries from one DB2 for i system to another. Once setup, this replication is handled automatically and transparently by the operating system. The entries that are replicated are placed in a journal on the target system that is available to be read by an application in the same way as on the source system.

You must have an understanding of how to setup and use remote journaling on an DB2 for i system to use this feature with Oracle GoldenGate. There are no special software requirements for either Oracle GoldenGate or the DB2 for i systems to use remote journaling.

This chapter contains the following sections:

- [Section 6.1, "Preparing to Use Remote Journals"](#)
- [Section 6.2, "Adding a Remote Journal"](#)

6.1 Preparing to Use Remote Journals

Before establishing the remote journal environment, complete the following steps:

1. Determine the extent of your remote journal network or environment.
2. *Library redirection* is the ability to allow the remote journal and associated journal receivers to reside in different libraries on the target system from the corresponding source journal and its associated journal receivers.

Determine what library redirection, if any, you will be using for the remote journals and associated journal receivers.

3. Ensure that all selected libraries exist on the target systems. You must consider whether or not library redirection will be used when adding the remote journal.
4. Create the appropriate local journal if it does not already exist.
5. Configure and activate the communications protocol you have chosen to use.
6. After you have configured the communications protocol, it must be active while you are using the remote journal function.

For example, if you are using the OptiConnect for IBM i bus transport method, then the OptiConnect for IBM i subsystem, QSOC, must be active. QSOC must be active for both the source system and the target system, and the appropriate controllers and devices must be varied on. If you are using a SNA communications transport, vary on the appropriate line, controller, and devices and ensure

subsystem QCMN is active on both systems. Start of change If you are using TCP/IP or Sockets IPv6, you must start TCP/IP by using the Start TCP/IP (STRTCP) command, including the distributed data management (DDM) servers. If you are using data port, you must configure a cluster, make sure that the cluster is active, and start the internet Daemon (inetd) server using the Start TCP/IP Server (STRTCPSVR) command. End of change

7. If one does not already exist, create the appropriate relational database (RDB) directory entry that will be used to define the communications protocol for the remote journal environment. When TCP communications are being used to connect to an independent disk pool, the RDB entry to the independent disk pool must have the Relational database value set to the target system's local RDB entry and the relational database alias value set to the independent disk pool's name.
8. Now you should be able to see the remote database connection by issuing the WRKRDBDIRE command.

Work with Relational Database Directory Entries

Position to

Type options, press Enter.

1=Add 2=Change 4=Remove 5=Display details 6=Print details

Remote

Option Entry Location Text

SYS1 system1

SYS2 system2

MYSYSTEM *LOCAL Entry added by system

Bottom

F3=Exit F5=Refresh F6=Print list F12=Cancel F22=Display entire field
(C) COPYRIGHT IBM CORP. 1980, 2007.

6.2 Adding a Remote Journal

Adding a remote journal creates a remote journal on a target system or independent disk pool and associates that remote journal with the journal on the source system. This occurs if this is the first time the remote journal is being established for a journal. The journal on the source system can be either a local or remote journal.

If a remote journal environment has previously been established, adding a remote journal reassociates the remote journal on the target system with the journal on the source system.

You can establish and associate a remote journal on a target system with a journal on the source system by one of the following methods:

- System i Navigator.
- Add the Remote Journal (QjoAddRemoteJournal) API on the source system.
- Add the Remote Journal (ADDRMTJRN) command on the source system.

6.2.1 What Happens During Add Remote Journal Processing?

The processing that takes place as part of adding a remote journal includes the following:

- A check is performed on the target system to verify that the user profile adding the remote journal exists. A user profile with the same name as the user profile which is adding a remote journal must exist on the target system. If the profile does not exist on the target system, then an exception is signaled, and the processing ends.
- A check is performed to verify that the target system has a library by the same name as the library for the journal on the source system. If the library does not exist on the target system, then an exception is signaled, and the processing ends.
- A check is performed on the target system to determine if a journal by the same qualified name as the journal on the source system already exists. If a journal already exists, it can be used for the remainder of the add remote journal processing if it meets the following criteria:
 1. It is a remote journal.
 2. It was previously associated with this same source journal or part of the same remote journal network.
 3. The type of the remote journal matches the specified remote journal type.
- If a journal was found, but does not meet the preceding criteria, then an exception is signaled, and the processing ends. Otherwise, the remote journal is used for the rest of the add remote journal processing.
- If no journal is found on the specified target system, then a remote journal is created on the target system. The new remote journal has the same configuration, authority, and audit characteristics of the source journal. The journal that is created has a journal type of *REMOTE.

When adding the remote journal, you must specify the type of remote journal to add. The remote journal type influences the library redirection rules and other operational characteristics for the journal.

6.2.2 Guidelines For Adding a Remote Journal

You should observe the following guidelines for adding a remote journal:

- You can only associate a remote journal with a single source journal.
 Note: The same remote journal can then have additional remote journals that are associated with it that are located on other target systems. This is the cascade configuration that is shown in Network configurations for remote journals.
- The remote journal will only have its attached receiver populated with journal entries that are replicated from the corresponding journal receiver on the source system. No journal entries can be directly deposited to a remote journal.
- A maximum of 255 remote journals can be associated with a single journal on a source system. This can be any combination of asynchronously maintained or synchronously maintained remote journals.

To Add a Remote Journal

The following is an example using the physical file QGPL/TESTPF setup to have remote journaling enabled to a second system.

1. Create the physical file.:

```
> CRTPF FILE(QGPL/TESTPF) RCDLEN(10)
File TESTPF created in library QGPL.
Member TESTPF added to file TESTPF in QGPL.
```

2. Create the local journal receiver and journals, and enable the journaling of the physical file created:

```
> crtjrnrcv jrnrcv(qgpl/jrcvrmt)
Journal receiver JRCVRMT created in library QGPL

> crtjrn jrn(qgpl/jrnrmnt) jrnrcv(qgpl/jrcvrmt) fixlenda(*job *usr *pgm
*sysseq)
Journal JRNRMNT created in library QGPL

strjrnpf file(qgpl/testpf) jrn(qgpl/testpf)
1 of 1 files have started journaling
```

3. Add the remote journal:

```
> addrmtjrn rdb(sys2) srcjrn(qgpl/JRNRMNT) rmtjrntype(*TYPE2)
Remote journal JRNRMNT in QGPL was added
```

4. Activate the remote journaling:

```
> chgrmtjrn rdb(sys2) srcjrn(qgpl/jrnrmnt) jrnstate(*active)
Remote journal JRNRMNT in library QGPL was activated
```

Uninstalling Oracle GoldenGate

This chapter contains instructions for removing Oracle GoldenGate from the system.

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.

This chapter contains the following sections:

- [Section 7.1, "Uninstalling Oracle GoldenGate from a Source DB2 for i System"](#)
- [Section 7.2, "Uninstalling Oracle GoldenGate from a Linux Remote System"](#)
- [Section 7.3, "Removing Oracle GoldenGate from a Windows Remote System"](#)

7.1 Uninstalling Oracle GoldenGate from a Source DB2 for i System

1. (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.

2. From the Oracle GoldenGate installation directory, run GGSCI.

3. Stop all Oracle GoldenGate processes.

```
STOP ER *
```

4. Stop the Manager process.

```
STOP MANAGER
```

5. 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.

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

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

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

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

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

```
clrlib library  
dltlib library
```

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

7.2 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.

7.3 Removing Oracle GoldenGate from a Windows Remote System

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

7.3.1 Removing Oracle GoldenGate from a Windows Cluster

1. Working from the node in the cluster that owns the Windows Cluster group that contains the Manager resource, run GGSCI and then stop any Extract and Replicat processes that are still running.
2. Use the Windows 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. Move the Windows Cluster group to the next node in the cluster, and repeat from 4.
8. Follow the instructions in [Removing Oracle GoldenGate from a Remote Windows System](#).

7.3.2 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. 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.

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

```
DELETE CHECKPOINTTABLE owner.table
```

10. 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.

This appendix includes the following sections:

- [Section A.1, "Oracle GoldenGate Programs and Utilities"](#)
- [Section A.2, "Oracle GoldenGate Subdirectories"](#)
- [Section A.3, "Other Oracle GoldenGate Files"](#)
- [Section A.4, "Oracle GoldenGate Checkpoint Table"](#)
- [Section A.5, "Oracle GoldenGate DB2 for i Objects"](#)

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.
convprm	Converts parameter files that do not use SQL-92 rules for quoted names and literals to updated parameter files that use SQL-92 rules. SQL-92 format for quoted object names and literals was introduced as the default with version 12c of Oracle GoldenGate.
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.

Table A–1 (Cont.) Oracle GoldenGate Installed Programs and Utilities

Program	Description
extract	Performs capture from database tables or transaction logs or receives transaction data from a vendor access module.
ggmxinstall	Oracle GoldenGate installation script for the SQL/MX database.
ggcmd	Associated program of ggsci. Launches and monitors external applications, such as the JAGENT of Oracle GoldenGate Monitor. Integrates those applications into the ggsci environment.
ggsci	User interface to Oracle GoldenGate for issuing commands and managing parameter files.
ggsmgr.jcl ggsmgr.proc ggsmgrst.jcl ggsmgrst.proc	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.
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.
oggerr	Manages Oracle GoldenGate error messages.
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.
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.

Table A–2 (Cont.) Oracle GoldenGate Installed Subdirectories

Directory	Description
dirdb	Contains the data store 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.
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>rep1.cpr</p>
dircrd	Contains credential store files.
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>
dirdmp	Contains trace, or dump, files that support the internal activity logging mechanism.
dirjar	Contains the Java executable files that support Oracle GoldenGate Monitor.

Table A–2 (Cont.) Oracle GoldenGate Installed Subdirectories

Directory	Description
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><i>mgr.pcm</i></p> <p><i>ext.pce</i></p>
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><i>defgen.prm</i></p> <p><i>finance.prm</i></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><i>fin2.rpt</i></p> <p><i>mgr4.rpt</i></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	<p>The default location for storing transaction data when the size exceeds the memory size that is allocated for the cache manager.</p> <p>Do not edit these files.</p>
dirwlt	Contains Oracle GoldenGate wallet files.
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
bcpfmt.tpl	Template for use with Replicat when creating a run file for the Microsoft BCP/DTS bulk-load utility.
bcrypt.txt	Blowfish encryption software license agreement.
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_cleartrace.sql	Script that removes the DDL trace file. (Oracle installations)
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_nopurgeRecyclebin.sql	Empty script file for use by Oracle GoldenGate support staff.
ddl_ora11.sql ddl_ora12.sql	Scripts that run programmatically as part of Oracle GoldenGate DDL support; do not run these scripts.
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 ddl_session1.sql	Supports the installation of the Oracle DDL objects. This script runs programmatically; do not run it manually.
ddl_setup.sql	Script that installs the Oracle GoldenGate DDL extraction and replication objects. (Oracle installations)
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_trace_off.sql ddl_trace_on.sql	Scripts that control whether DDL tracing is on or off.

Table A-3 (Cont.) Other Oracle GoldenGate Installed Files

Component	Description
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.
.dump 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.
icudtxx.dll icuinx.dll icuucxx.dll	Windows shared libraries for International Components for Unicode, where xx is the currently used version.
jagent.bat	Windows batch file for the Java Agent for Oracle GoldenGate Monitor.
jagent.log jagentjni.log	Log files for the Oracle GoldenGate Monitor Agent.
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)

Table A–3 (Cont.) Other Oracle GoldenGate Installed Files

Component	Description
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.
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	
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). For a description of this table, see *Administering Oracle GoldenGate for Windows and UNIX*.

A.5 Oracle GoldenGate DB2 for i Objects

This section lists the items that are installed in the Oracle GoldenGate library.

Table A–4 *Oracle GoldenGate DB2 for i Objects*

Name	Type	Description
OGGPRCCMD	*PGM	Command processing program
OGGPRCJRN	*SRVPGM	Extract Service Program
OGGRCVXXXX	*JRNRCV	Journal Receivers
OGGJRN	*JRN	Oracle GoldenGate journal
DEFGEN	*CMD	DEFGEN command
EXTRACT	*CMD	Extract command
GGSCI	*CMD	GGSCI command
KEYGEN	*CMD	KEYGEN command
LOGDUMP	*CMD	Logdump command
MGR	*CMD	Manager command
REPLICAT	*CMD	Replicat command
OGGDIR	*DTAARA	Installation directory data area