

Oracle® Database Gateway

Installation and Configuration Guide

11g Release 1 (11.1) for AIX 5L Based Systems (64-Bit), HP-UX
PA-RISC (64-Bit), HP-UX Itanium, Solaris Operating System
(SPARC 64-Bit), Linux x86, and Linux x86-64

B31042-04

August 2008

Oracle Database Gateway Installation and Configuration Guide, 11g Release 1 (11.1) for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), HP-UX Itanium, Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64

B31042-04

Copyright © 2006, 2008, Oracle. All rights reserved.

Primary Author: Maitreyee Chaliha

Contributor: Vira Goorah, Den Raphaely, Govind Lakkoju, Peter Wong, Juan Pablo Ahues-Vasquez, Peter Castro, and Charles Benet

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Preface	xi
Intended Audience.....	xi
Documentation Accessibility	xi
Related Documents	xii
Conventions	xii
Part I Overview of the Oracle Database Gateway Installation	
1 Overview of the Oracle Database Gateway Installation	
Gateway Installation Configurations.....	1-1
Gateway Installation Methods	1-1
Interactive Installation Method.....	1-1
Automated Installation Method Using Response Files.....	1-2
Installation Considerations.....	1-2
Release Notes.....	1-2
Hardware and Software Certification.....	1-2
Multiple Oracle Homes Support.....	1-3
Upgrades	1-3
Accessing the Installation Software	1-3
Downloading Oracle Software from the OTN Web Site	1-3
Copying the Oracle Software to a Hard Disk	1-4
Running the Oracle Universal Installer.....	1-5
Part II Installing and Configuring Oracle Database Gateway for Sybase	
2 Installing Oracle Database Gateway for Sybase	
System Requirements for Oracle Database Gateway for Sybase.....	2-1
Hardware Requirements.....	2-1
Software Requirements	2-3
Step Through the Oracle Universal Installer.....	2-5
3 Configuring Oracle Database Gateway for Sybase	
Configure the Gateway Initialization Parameter File.....	3-1
Choose a System Identifier for the Gateway.....	3-1

Customize the Initialization Parameter File.....	3-2
Configure Oracle Net for the Gateway	3-2
Configure Oracle Net Listener for the Gateway.....	3-2
Stop and Start the Oracle Net Listener for the Gateway	3-5
Configure the Oracle Database for Gateway Access.....	3-6
Configuring tnsnames.ora	3-6
Create Database Links.....	3-7
Configure Two-Phase Commit	3-8
Create a Recovery Account and Password.....	3-8
Create the Transaction Log Table	3-9
Create Sybase Views for Data Dictionary Support	3-9
Encrypt Gateway Initialization Parameter Values.....	3-10
Configure the Gateway to Access Multiple Sybase Databases.....	3-10
Multiple Sybase Databases Example: Configuring the Gateway.....	3-10
Multiple Sybase Databases Example: Configuring Oracle Net Listener.....	3-11
Multiple Sybase Databases Example: Stopping and Starting the Oracle Net Listener.....	3-11
Multiple Sybase Databases Example: Configuring Oracle Database for Gateway Access ..	3-12
Multiple Sybase Databases Example: Accessing Sybase Data.....	3-12

Part III Installing and Configuring Oracle Database Gateway for Informix

4 Installing Oracle Database Gateway for Informix

System Requirements for Oracle Database Gateway for Informix	4-1
Hardware Requirements.....	4-1
Software Requirements	4-3
Step Through the Oracle Universal Installer.....	4-5

5 Configuring Oracle Database Gateway for Informix

Configure the Gateway Initialization Parameter File.....	5-1
Choose a System Identifier for the Gateway.....	5-1
Customize the Initialization Parameter File.....	5-1
Configure Oracle Net for the Gateway	5-2
Configure Oracle Net Listener for the Gateway.....	5-2
Stop and Start the Oracle Net Listener for the Gateway	5-5
Configure the Oracle Database for Gateway Access.....	5-6
Configuring tnsnames.ora	5-6
Create Database Links.....	5-7
Configure Two-Phase Commit	5-8
Create a Recovery Account and Password.....	5-8
Create the Transaction Log Table	5-9
Encrypt Gateway Initialization Parameter Values.....	5-10
Configure the Gateway to Access Multiple Informix Databases	5-10
Multiple Informix Databases Example: Configuring the Gateway	5-10
Multiple Informix Databases Example: Configuring Oracle Net Listener	5-11
Multiple Informix Databases Example: Stopping and Starting the Oracle Net Listener	5-11

Multiple Informix Databases Example: Configuring Oracle Database for Gateway Access	5-12
Multiple Informix Databases Example: Accessing Informix Data.....	5-12

Part IV Installing and Configuring Oracle Database Gateway for Teradata

6 Installing Oracle Database Gateway for Teradata

System Requirements for Oracle Database Gateway for Teradata	6-1
Hardware Requirements.....	6-1
Software Requirements	6-3
Step Through the Oracle Universal Installer	6-4

7 Configuring Oracle Database Gateway for Teradata

Configure the Gateway Initialization Parameter File	7-1
Choose a System Identifier for the Gateway.....	7-1
Customize the Initialization Parameter File.....	7-1
Configure Oracle Net for the Gateway	7-2
Configure Oracle Net Listener for the Gateway.....	7-2
Stop and Start the Oracle Net Listener for the Gateway	7-5
Configure the Oracle Database for Gateway Access	7-6
Configuring tnsnames.ora	7-6
Create Database Links	7-7
Configure Two-Phase Commit	7-8
Create a Recovery Account and Password.....	7-8
Create the Transaction Log Table	7-9
Encrypt Gateway Initialization Parameter Values	7-9
Configure the Gateway to Access Multiple Teradata Databases	7-10
Multiple Teradata Databases Example: Configuring the Gateway	7-10
Multiple Teradata Databases Example: Configuring Oracle Net Listener	7-11
Multiple Teradata Databases Example: Stopping and Starting the Oracle Net Listener	7-11
Multiple Teradata Databases Example: Configuring Oracle Database for Gateway Access	7-11
Multiple Teradata Databases Example: Accessing Teradata Data	7-12

Part V Installing and Configuring Oracle Database Gateway for SQL Server

8 Installing Oracle Database Gateway for SQL Server

System Requirements for Oracle Database Gateway for SQL Server	8-1
Hardware Requirements.....	8-1
Software Requirements	8-3
Step Through the Oracle Universal Installer	8-5

9 Configuring Oracle Database Gateway for SQL Server

Configure the Gateway Initialization Parameter File	9-1
Choose a System Identifier for the Gateway.....	9-1
Customize the Initialization Parameter File.....	9-2

Configure Oracle Net for the Gateway	9-2
Configure Oracle Net Listener for the Gateway.....	9-3
Stop and Start the Oracle Net Listener for the Gateway	9-5
Configure the Oracle Database for Gateway Access.....	9-6
Configuring tnsnames.ora	9-6
Create Database Links.....	9-7
Configure Two-Phase Commit	9-8
Create a Recovery Account and Password.....	9-8
Create the Transaction Log Table	9-9
Create SQL Server Views for Data Dictionary Support	9-10
Encrypt Gateway Initialization Parameter Values.....	9-10
Configure the Gateway to Access Multiple SQL Server Databases.....	9-10
Multiple SQL Server Databases Example: Configuring the Gateway.....	9-11
Multiple SQL Server Databases Example: Configuring Oracle Net Listener.....	9-11
Multiple SQL Server Databases Example: Stopping and Starting the Oracle Net Listener	9-12
Multiple SQL Server Databases Example: Configuring Oracle Database for Gateway Access	9-12
Multiple SQL Server Databases Example: Accessing SQL Server Data.....	9-13

Part VI Installing and Configuring Oracle Database Gateway for ODBC

10 Installing Oracle Database Gateway for ODBC

System Requirements for Oracle Database Gateway for ODBC.....	10-1
Hardware Requirements.....	10-1
Software Requirements	10-3
Step Through the Oracle Universal Installer.....	10-5

11 Configuring Oracle Database Gateway for ODBC

Configure the Gateway Initialization Parameter File.....	11-1
Create the Initialization Parameter File	11-1
Set the Initialization Parameter Values.....	11-2
Configure Oracle Net for the Gateway	11-3
Configure Oracle Net Listener for the Gateway.....	11-3
Stop and Start the Oracle Net Listener for the Gateway	11-5
Configure the Oracle Database for Gateway Access.....	11-6
Configuring tnsnames.ora	11-6
Create Database Links.....	11-7
Encrypt Gateway Initialization Parameter Values.....	11-8
Configure the Gateway to Access Multiple ODBC Data Sources	11-8
Multiple ODBC Data Sources Example: Configuring the Gateway	11-8
Multiple ODBC Data Sources Example: Configuring Oracle Net Listener	11-9
Multiple ODBC Data Sources Example: Stopping and Starting the Oracle Net Listener	11-9
Multiple ODBC Data Sources Example: Configuring Oracle Database for Gateway Access.....	11-10
Multiple ODBC Data Sources Example: Accessing ODBC Data.....	11-10

Part VII Installing and Configuring Oracle Database Gateway for

DRDA

12 Installing Oracle Database Gateway for DRDA

System Requirements for Oracle Database Gateway for DRDA	12-1
Hardware Requirements	12-1
Software Requirements	12-3
Step through the Oracle Universal Installer	12-4

13 Configuring the DRDA Server

Configuring the DRDA Server for DB2/OS390	13-1
Configuring the DRDA Server for DB2/400	13-3
Configuring the DRDA Server for DB2/UDB (Universal Database)	13-3

14 Configuring Oracle Database Gateway for DRDA

Configure the Gateway Initialization Parameter File	14-1
Choose a System Identifier for the Gateway	14-2
Customize the Initialization Parameter File	14-2
Configure Oracle Net for the Gateway	14-3
Configure Oracle Net Listener for the Gateway	14-3
Stop and Start the Oracle Net Listener for the Gateway	14-4
Configure Two-Phase Commit	14-5
Bind the DRDA Gateway Package	14-6
DRDA Gateway Package Binding Considerations	14-6
DRDA Gateway Package Binding Steps	14-7
Create Tables and Views for Data Dictionary Support	14-7
Grant Authority to the DRDA Package	14-8
Configure the Oracle Database for Gateway Access	14-9
Configuring tnsnames.ora	14-9
Create Database Links	14-10
Configure the Gateway to Access Multiple DRDA Databases	14-10
Multiple DRDA Databases Example: Configuring the Gateway	14-11
Multiple DRDA Databases Example: Configuring Oracle Net Listener	14-11
Multiple DRDA Databases Example: Stopping and Starting the Oracle Net Listener	14-12
Multiple Databases Example: Configuring Oracle Database for Gateway Access	14-12
Multiple DRDA Databases Example: Accessing DB2 Data	14-13

15 Security Considerations

Security Overview	15-1
Authenticating Application Logons	15-1
Defining and Controlling Database Links	15-2
Link Accessibility	15-2
Links and CONNECT Clauses	15-2
Processing Inbound Connections	15-2
User ID Mapping	15-3
Passwords in the Gateway Initialization File	15-4

16 Migration From Previous Releases

Install the New Release.....	16-1
Copy the Gateway Initialization Parameter File.....	16-1
Update the Initialization Parameters.....	16-1
Changed Parameters.....	16-1
Obsolete Parameters	16-2
Bind Gateway Package.....	16-2
Install/Upgrade Data Dictionary Views.....	16-2

Part VIII Removing Oracle Database Gateway

17 Removing Oracle Database Gateway

Part IX Appendixes

A Using Response Files for Noninteractive Installation

Introduction.....	A-1
Installation Overview	A-2
Creating the oraInst.loc File	A-2
Preparing a Response File	A-3
Editing a Response File Template.....	A-3
Recording a Response File	A-4
Running Oracle Universal Installer in Silent or Suppressed Mode	A-5

B Oracle Database Gateway Troubleshooting

Verify Requirements.....	B-1
What to Do If an Installation Error Occurs	B-1
Reviewing the Log of an Installation Session	B-2
Troubleshooting Configuration Assistants	B-2
Configuration Assistant Failure	B-3
Fatal Errors.....	B-3
Silent-Mode Response File Error Handling.....	B-3
Cleaning Up After a Failed Installation.....	B-4

C Initialization Parameters

Initialization Parameter File Syntax	C-1
Oracle Database Gateway for Sybase Initialization Parameters	C-2
Oracle Database Gateway for Informix Initialization Parameters	C-3
Oracle Database Gateway for Teradata Initialization Parameters.....	C-4
Oracle Database Gateway for SQL Server Initialization Parameters	C-4
Oracle Database Gateway for ODBC Initialization Parameters	C-5
Oracle Database Gateway for DRDA Initialization Parameters.....	C-6
Initialization Parameter Description	C-7
HS_CALL_NAME	C-7
HS_DB_DOMAIN	C-8

HS_DB_INTERNAL_NAME	C-8
HS_DB_NAME	C-8
HS_DESCRIBE_CACHE_HWM	C-8
HS_LANGUAGE	C-9
HS_LONG_PIECE_TRANSFER_SIZE	C-10
HS_OPEN_CURSORS	C-10
HS_RPC_FETCH_REBLOCKING	C-10
HS_RPC_FETCH_SIZE	C-11
HS_TIME_ZONE	C-11
HS_TRANSACTION_MODEL	C-11
IFILE	C-12
HS_FDS_CONNECT_INFO	C-12
HS_FDS_DEFAULT_OWNER	C-13
HS_FDS_PROC_IS_FUNC.....	C-14
HS_FDS_RECOVERY_ACCOUNT	C-14
HS_FDS_RECOVERY_PWD.....	C-14
HS_FDS_RESULTSET_SUPPORT	C-15
HS_FDS_TRACE_LEVEL.....	C-15
HS_FDS_TRANSACTION_LOG	C-15
HS_FDS_SHAREABLE_NAME	C-16
HS_FDS_REPORT_REAL_AS_DOUBLE	C-16
HS_FDS_FETCH_ROWS.....	C-16
DRDA_CACHE_TABLE_DESC.....	C-16
DRDA_CAPABILITY.....	C-17
DRDA_CODEPAGE_MAP.....	C-17
DRDA_COMM_BUFLLEN	C-17
DRDA_CONNECT_PARM	C-17
DRDA_DEFAULT_CCSID.....	C-18
DRDA_DESCRIBE_TABLE	C-18
DRDA_DISABLE_CALL.....	C-19
DRDA_FLUSH_CACHE.....	C-19
DRDA_GRAPHIC_CHAR_SIZE	C-19
DRDA_GRAPHIC_PAD_SIZE.....	C-20
DRDA_GRAPHIC_LIT_CHECK	C-20
DRDA_GRAPHIC_TO_MBCS	C-20
DRDA_ISOLATION_LEVEL.....	C-20
DRDA_LOCAL_NODE_NAME	C-21
DRDA_MBCS_TO_GRAPHIC	C-21
DRDA_OPTIMIZE_QUERY	C-22
DRDA_PACKAGE_COLLID.....	C-22
DRDA_PACKAGE_CONSTOKEN	C-22
DRDA_PACKAGE_NAME	C-23
DRDA_PACKAGE_OWNER	C-23
DRDA_PACKAGE_SECTIONS.....	C-23
DRDA_READ_ONLY	C-24
DRDA_RECOVERY_PASSWORD	C-24
DRDA_RECOVERY_USERID	C-24

DRDA_REMOTE_DB_NAME.....	C-25
FDS_CLASS.....	C-25
HS_NLS_NCHAR.....	C-25
LOG_DESTINATION.....	C-26
ORA_MAX_DATE.....	C-26
ORA_NLS11.....	C-26
ORACLE_DRDA_TCTL.....	C-26
ORACLE_DRDA_TRACE.....	C-27
TRACE_LEVEL.....	C-27
HS_NLS_DATE_FORMAT.....	C-27
HS_NLS_DATE_LANGUAGE.....	C-28
HS_NLS_NUMERIC_CHARACTER.....	C-28

D Configuration Worksheet for DRDA

E Globalization Support for DRDA

Overview of Globalization Support Interactions.....	E-1
Client and Oracle Database Configuration.....	E-3
Gateway Language Interaction with DRDA Server.....	E-4
Gateway Configuration.....	E-4
Globalization Support Parameters in the Gateway Initialization File.....	E-4
Gateway Codepage Map Facility.....	E-6
Multi-Byte and Double-Byte Support in the Gateway.....	E-8
Message Availability.....	E-10
Example of Globalization Support Configuration.....	E-10

Index

Preface

This guide describes how to install and configure Oracle Database Gateway for Sybase, Informix, Teradata, SQL Server, ODBC, and DRDA on UNIX based platforms.

Intended Audience

This manual is intended for users responsible for installing Oracle Database Gateways on UNIX based platforms.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, 7 days a week. For TTY support, call 800.446.2398. Outside the United States, call +1.407.458.2479.

Related Documents

For more information, see the following documents:

- *Oracle Database Gateway for Sybase User's Guide*
- *Oracle Database Gateway for Informix User's Guide*
- *Oracle Database Gateway for Teradata User's Guide*
- *Oracle Database Gateway for Microsoft SQL Server User's Guide*
- *Oracle Database Gateway for DRDA User's Guide*
- *Oracle Database New Features Guide*
- *Oracle Call Interface Programmer's Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Database Advanced Application Developer's Guide*
- *Oracle Database Concepts*
- *Oracle Database Performance Tuning Guide*
- *Oracle Database Error Messages*
- *Oracle Database Globalization Support Guide*
- *Oracle Database Reference*
- *Oracle Database SQL Language Reference*
- *Oracle Database Net Services Administrator's Guide*
- *SQL*Plus User's Guide and Reference*
- *Oracle Database Heterogeneous Connectivity Administrator's Guide*
- *Oracle Database Security Guide*

Conventions

The following typographic conventions are used in this manual:

Convention	Meaning
bold	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary
<i>italics</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter, directory names, usernames, pathnames, and filenames.
UPPERCASE	Uppercase letters indicate Structured Query Language (SQL) reserved words, initialization parameters, and environment variables.
[text]	Brackets are used in syntax statements for optional elements.
[text text]	Vertical bar inside brackets is used in syntax statements to imply choice among optional elements.
{text text}	Vertical bar inside braces is used in syntax statements to imply choice among mandatory elements.

Part I

Overview of the Oracle Database Gateway Installation

Part I contains the following chapter:

- [Chapter 1, "Overview of the Oracle Database Gateway Installation"](#)

Overview of the Oracle Database Gateway Installation

This chapter describes issues that you should consider before installing the Oracle Database Gateways. It includes information about the following topics:

- [Gateway Installation Configurations](#)
- [Gateway Installation Methods](#)
- [Installation Considerations](#)
- [Upgrades](#)
- [Accessing the Installation Software](#)
- [Running the Oracle Universal Installer](#)

Gateway Installation Configurations

You can install Oracle Database Gateway in either of the following configurations:

- On the same computer as an existing Oracle database but in a different Oracle home.
- On a system with no Oracle database.
- On the same computer as the Oracle database, and in the same Oracle home directory. Note that in this case, the Oracle database and the gateway must be at the same release level.

Gateway Installation Methods

Following are the installation methods to install Oracle Database Gateways:

- [Interactive Installation Method](#)
- [Automated Installation Method Using Response Files](#)

Interactive Installation Method

When you use the interactive method to install Oracle Database Gateways, Oracle Universal Installer displays a series of screens that enable you to specify all of the required information.

Automated Installation Method Using Response Files

By creating a response file and specifying this file when you start Oracle Universal Installer, you can automate some or all of the Oracle Database Gateway installation. These automated installation methods are useful if you need to perform multiple installations on similarly configured systems or if the system where you want to install the software does not have X Window system software installed.

When you use a response file, you can run Oracle Universal Installer in the following modes, depending on whether you specify all of the required information or not:

- Silent Mode

Oracle Universal Installer runs in silent mode if you use a response file that specifies all required information. None of the Oracle Universal Installer screens are displayed.

- Suppressed Mode

Oracle Universal Installer runs in suppressed mode if you do not specify all required information in the response file. Oracle Universal Installer displays only the screens that prompt for the information that you did not specify.

For more information about these modes and about how to complete an installation using response files, refer to [Appendix A, "Using Response Files for Noninteractive Installation"](#).

Installation Considerations

This section contains information that you should consider before installing this product. They are:

- [Release Notes](#)
- [Hardware and Software Certification](#)
- [Multiple Oracle Homes Support](#)

Release Notes

Read the release notes for the product before installing it. The release notes are available on the Oracle Database 11g Release 1 (11.1) installation media. The latest version of the release notes is also available on the OTN Web site:

<http://www.oracle.com/technology/documentation/index.html>

Hardware and Software Certification

The platform-specific hardware and software requirements included in this installation guide were current at the time this guide was published. However, because new platforms and operating system software versions might be certified after this guide is published, review the certification matrix on the *OracleMetaLink* Web site for the most up-to-date list of certified hardware platforms and operating system versions. The *OracleMetaLink* Web site is available at the following Web site:

<https://metalink.oracle.com>

If you do not have a current Oracle Support Services contract, then you can access the same information at the following Web site:

<http://www.oracle.com/technology/support/metalink/content.html>

Multiple Oracle Homes Support

This product supports multiple Oracle homes. This means that you can install this release or previous releases of the software more than once on the same system, in different Oracle home directories.

Installing the Software on a System with an Existing Oracle Installation

You must install this product into a new Oracle home directory. You cannot install products from one release of Oracle Database Gateways into an Oracle home directory of a different release. For example, you cannot install release 11.1 software into an existing Oracle 10gR2 Oracle home directory. If you attempt to install this release into an Oracle home directory that contains software from an earlier Oracle release, the installation fails.

You can install this release more than once on the same system if each installation is installed in a separate Oracle home directory.

Upgrades

Upgrades are not supported for Oracle Database Gateways.

Accessing the Installation Software

You can access the Oracle Database Gateway software by using one of the following methods:

- Download the software from OTN. Refer to [Downloading Oracle Software from the OTN Web Site](#).
- Copy the software to a hard disk. Refer to [Copying the Oracle Software to a Hard Disk](#)

Downloading Oracle Software from the OTN Web Site

This section describes how to download the installation archive files and extract them on your hard disk. It contains the following topics:

- [Downloading the Installation Archive Files](#)
- [Extracting the Installation Files](#)

Downloading the Installation Archive Files

To download the installation archive files from OTN:

1. Use any browser to access the software download page on OTN:

<http://www.oracle.com/technology/software/>

2. Navigate to the download page for the product that you want to install.
3. Select a file system with enough free space to store and expand the archive files.

In most cases, the available disk space must be at least twice the size of the archive files.

4. On the file system that you selected in step 3, create a directory, for example, *gateway*, to hold the installation archive files.
5. Download the installation archive files to the directory that you created in step 4.

6. Verify that the files you downloaded are the same size as the corresponding files on OTN.

Extracting the Installation Files

To extract the installation archive files, perform the following steps:

1. If necessary, change directory to the directory that contains the downloaded installation archive files.
2. To uncompress each file, enter a command similar to the following:

```
$ gunzip filename.cpio.gz
```

This command creates files with names similar to the following:

```
filename.cpio
```

3. To extract the installation files, enter a command similar to the following:

```
$ cpio -idmv < filename.cpio
```

Note: Refer to the download page for information about the correct options to use with the `cpio` command.

Some browsers uncompress files while downloading them, but leave the `.gz` file extension. If these steps do not work, remove the `.gz` extension from the files and repeat step 3.

For each file, this command creates a subdirectory named `Diskn`, where `n` is either 1 or the disk number identified in the file name.

Copying the Oracle Software to a Hard Disk

Before installing Oracle Database Gateway, you might want to copy the software to the hard disk. This enables the installation process to run a bit faster. Before copying the installation media content to the hard disk, you must mount the installation media. The following section describes how to mount discs and copy their content to the hard disk.

Mounting Disc

On most Solaris Operating Systems, the disc mounts automatically when you insert it into the disc drive. If the disc does not mount automatically, follow these steps to mount it:

1. Switch user to root

```
$ su - root
```

2. If necessary, enter a command similar to one of the following to eject the currently mounted disc, then remove it from the drive:

Solaris (SPARC):

```
# eject
```

AIX:

```
# umount /cdrom
```

HP-UX PA-RISC:

```
# /usr/sbin/umount /SD_CDROM
```

In these examples, `/cdrom` and `/SD_CDROM` are the mount point directories for the disc drive.

3. Insert the appropriate disc into the disc drive.
4. To verify that the disc mounted automatically, enter a command similar to the following depending on your platform:

Solaris (SPARC):

```
# ls /cdrom/cdrom0
```

5. If this command fails to display the contents of the disc, enter a command similar to the following to mount it, depending on your platform:

Solaris (SPARC):

```
# /usr/sbin/mount -r -F hsfs /dev/dsk/cxydzs2 /cdrom
```

In this example, `/cdrom` is the disc mount point directory and `/dev/dsk/cxydzs2` is the device name for the disc device, for example `/dev/dsk/c0t2d0s2`.

6. If Oracle Universal Installer is displaying the Disk Location dialog box, enter the disc mount point directory path, for example:

```
/mnt/cdrom
```

Copying the Oracle Database Gateway Software to a Hard Disk

To copy the contents of the media to a hard disk:

1. Create a directory on the hard disk to hold the Oracle Database Gateway software:

```
$ mkdir gateway
```

2. Change directory to the directory you created in step 1:

```
$ cd gateway
```

3. Copy the contents of the mounted disc to the new directory as follows:

```
$ cp -R /directory_path gateway
```

In this example, `/directory_path` is the installation media mount point directory. The mount point directory is `/cdrom`.

Running the Oracle Universal Installer

Start the Installer and install the software, as follows:

1. If you are installing the software from disc, mount the appropriate disc if it is not already mounted

Some platforms automatically mount discs when you insert them into the drive.

2. If necessary, log in as the Oracle software owner user (`oracle`) and set the `DISPLAY` environment variable.

3. To start the Installer, enter the following commands where *directory_path* is the CD-ROM mount point directory, the path of the `tg` directory on the DVD-ROM, or the directory path of the software on the hard disk.

```
$ /directory_path/runInstaller
```

4. Use the following guidelines to complete the installation:
 - Follow the instruction displayed in the Installer window. If you need additional information, click **Help**.
 - When the Installer prompts you to run a script with `root` privileges, enter a command similar to the following in a terminal where you are logged in as the root user, then click **Continue** or **OK**:

```
# /script_path/script_name
```

- If you encounter errors while installing or linking the software, then see [Appendix B, "Oracle Database Gateway Troubleshooting"](#) for information about troubleshooting.
5. When the installation is complete, click **Exit**, then click **Yes** to exit from the Installer.

Part II

Installing and Configuring Oracle Database Gateway for Sybase

Part II, "Installing and Configuring Oracle Database Gateway for Sybase" describes how to install and configure Oracle Database Gateway for Sybase on UNIX based platforms.

It contains the following chapters:

- Chapter 2, "Installing Oracle Database Gateway for Sybase"
- Chapter 3, "Configuring Oracle Database Gateway for Sybase"

Installing Oracle Database Gateway for Sybase

This chapter provides information about the hardware and software requirements and the installation procedure for Oracle Database Gateway for Sybase.

To install the gateway, follow these steps:

1. Ensure that the system meets all of the hardware and software requirements specified in "[System Requirements for Oracle Database Gateway for Sybase](#)" on page 2-1

2. Run the Oracle Universal Installer.

See "[Step Through the Oracle Universal Installer](#)" on page 2-5 for more information about running the Oracle Universal Installer

Oracle Universal Installer is a menu-driven utility that guides you through the installation of the gateway by prompting you with action items. The action items and the sequence in which they appear depend on your platform.

See [Table 2-3](#) for a description of the installation procedure of Oracle Database Gateway for Sybase

System Requirements for Oracle Database Gateway for Sybase

This section provides information about the hardware and software requirements for the gateway. It contains the following sections:

- "[Hardware Requirements](#)" on page 2-1
- "[Software Requirements](#)" on page 2-3

Hardware Requirements

[Table 2-1](#) shows the minimum hardware requirements for Oracle Database Gateway for Sybase.

Table 2–1 Hardware requirements for Oracle Database Gateway for Sybase

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for HP-UX Itanium	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Temporary Disk Space	400 MB	400 MB	400 MB	400 MB	400 MB	400 MB
Disk Space	1.5 GB	1.5 GB	1.5 GB	750 MB	750 MB	750 MB
Physical Memory*	512 MB	512 MB	512 MB	512 MB	512 MB	512 MB
Swap Space	1 GB	1 GB	1 GB	1 GB	1 GB	1 GB
Processor	IBM RS/6000 AIX-Based System Processor	HP 9000 Series 700 or 800 processor for hp-ux 11.0	HP Itanium processor for hp-ux 11	Sun Solaris Operating System (SPARC) Processor	x86	x86_64

* The minimum swap space is 1 GB (or twice the size of RAM). On systems with 2 GB or more of RAM, the swap space can be between one and two times the size of RAM. On AIX systems with 1 GB or more of memory, do not increase the swap space more than 2 GB.

Checking the Hardware Requirements

To ensure that the system meets the minimum requirements, follow these steps:

1. To determine the physical RAM size, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lstat -E -l sys0 -a realmem
HP-UX PA-RISC	# /usr/sbin/dmesg grep "Physical:"
HP-UX Itanium	# /usr/contrib/bin/machinfo grep -i Memory
Solaris (SPARC)	# /usr/sbin/prtconf grep "Memory size"
Linux x86	# grep MemTotal /proc/meminfo
Linux x86 64 bit	# grep MemTotal /proc/meminfo

If the size of the physical RAM installed in the system is less than the required size, you must install more memory before continuing.

2. To determine the size of the configured swap space, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lspv -a
HP-UX PA-RISC	# /usr/sbin/swapinfo -a
HP-UX Itanium	# /usr/sbin/swapinfo -a
Solaris (SPARC)	# /usr/sbin/swap -s
Linux x86	# grep SwapTotal /proc/meminfo
Linux x86 64 bit	# grep SwapTotal /proc/meminfo

If necessary, see your operating system documentation for information about how to configure additional swap space.

3. To determine the amount of disk space available in the `/tmp` directory enter the following commands:

Operating System	Command
AIX	# <code>df -k /tmp</code>
HP-UX PA-RISC	# <code>df -k /tmp</code>
HP-UX Itanium	# <code>bdf /tmp</code>
Solaris (SPARC)	# <code>df -k /tmp</code>
Linux x86	# <code>df -k /tmp</code>
Linux x86 64 bit	# <code>df -k /tmp</code>

4. To determine the amount of disk space available on the system enter the following commands:

Operating System	Command
AIX	# <code>df -k</code>
HP-UX PA-RISC	# <code>df -k</code>
HP-UX Itanium	# <code>bdf</code>
Solaris (SPARC)	# <code>df -k</code>
Linux x86	# <code>df -k</code>
Linux x86 64 bit	# <code>df -k</code>

Software Requirements

The following section describes the minimum software requirements for Oracle Database Gateway for Sybase.

Operating System

Table 2–2 shows the minimum operating system version required for Oracle Database Gateway for Sybase. If your operating system is lower than the minimum requirements, upgrade your operating system to meet the specified levels.

Table 2–2 Operating Systems version for Oracle Database Gateway for Sybase

Operating System	Version
AIX	AIX 5L version 5.3, Maintenance level 02 or higher
HP-UX PA-RISC	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
HP-UX Itanium	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
Solaris (SPARC)	Solaris 9 Update 6 or higher or Solaris 10
Linux x86 Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 Suse	SUSE Linux Enterprise Server 10.0

Table 2–2 (Cont.) Operating Systems version for Oracle Database Gateway for Sybase

Operating System	Version
Linux x86 64 bit Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 64 bit Suse	SUSE Linux Enterprise Server 10.0
Oracle Enterprise Linux x86	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0
Oracle Enterprise Linux x86 64 bit	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0

Checking the Software Requirements

To ensure that the system meets the minimum requirements, follow these steps:

- To determine which version of AIX is installed, enter the following command:

```
# oslevel -r
```
- To determine which version of HP-UX PA-RISC is installed, enter the following command:

```
# uname -a
```
- To determine which version of HP-UX Itanium is installed, enter the following command:

```
# uname -a
```
- To determine which version of Solaris Operating System (SPARC) is installed, enter the following command:

```
# uname -r
```
- To determine which distribution and version of Linux x86 is installed, enter the following command:

```
# cat /etc/issue
```
- To determine which distribution and version of Linux x86 64 bit is installed, enter the following command:

```
# cat /proc/version
```

Certified Configuration

The gateway supports Sybase Adaptive Server. For the latest versions supported refer to the OTN Web site:

<http://www.oracle.com/technology/products/gateways/pdf/certmatrix10g.pdf>

Step Through the Oracle Universal Installer

Table 2–3 describes the installation procedure for Oracle Database Gateway for Sybase.

Table 2–3 The Oracle Universal Installer: Steps for Installing the Gateway

Screen	Response
Oracle Universal Installer: Welcome	Click Next .
Oracle Universal Installer: File Locations	<p>The Source section of the screen is where you specify the source location that the Oracle Universal Installer must use to install the Oracle Database Gateway for Sybase. You need not edit the file specification in the Path field. The default setting for this field points to the installer file on your Oracle Database Gateway installation media.</p> <p>The Path field in the Destination section of the File Locations screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to <i>ORACLE_HOME</i>. After you set the fields in the File Locations screen as necessary, click Next to continue. After loading the necessary information from the installation media, the Oracle Universal Installer displays the Available Products screen.</p>
Oracle Universal Installer: Available Product Components	<p>a. Select Oracle Database Gateway for Sybase 11.1.0.6.0.</p> <p>b. Click Next.</p>
Oracle Database Gateway for Sybase	<p>Sybase Database Server Host Name - Specify the host name of the machine hosting the Sybase database server.</p> <p>Sybase Database Server Port number - Specify the port number of the Sybase database server</p> <p>Sybase Database Name - Specify the Sybase database name</p> <p>Click Next to continue.</p>
Oracle Universal Installer: Summary	The Installation Summary screen enables you to review a tree list of options and components for this installation. Click Install to start installation.
Oracle Net Configuration Assistant: Welcome	Click Cancel
Oracle Net Configuration Assistant:	Click Yes
Oracle Universal Installer: Configuration Tools	Click Exit
Exit	The final screen of the Oracle Universal Installer is the End of Installation screen. Click Exit to exit the installer.

Configuring Oracle Database Gateway for Sybase

After installing the gateway, perform the following tasks to configure Oracle Database Gateway for Sybase:

1. [Configure the Gateway Initialization Parameter File](#)
2. [Configure Oracle Net for the Gateway](#)
3. [Configure the Oracle Database for Gateway Access](#)
4. [Create Database Links](#)
5. [Configure Two-Phase Commit](#)
6. [Create Sybase Views for Data Dictionary Support](#)
7. [Encrypt Gateway Initialization Parameter Values](#)
8. [Configure the Gateway to Access Multiple Sybase Databases](#)

Configure the Gateway Initialization Parameter File

Perform the following tasks to configure the gateway initialization parameter file.

1. [Choose a System Identifier for the Gateway](#)
2. [Customize the Initialization Parameter File](#)

Choose a System Identifier for the Gateway

The gateway system identifier (SID) is an alphanumeric character string that identifies a gateway instance. You need one gateway instance, and therefore one gateway SID, for each Sybase database you are accessing. The SID is used as part of the file name for the initialization parameter file. The default SID is `dg4sybs`.

You can define a gateway SID, but using the default of `dg4sybs` is easier because you do not need to change the initialization parameter file name. However, if you want to access two Sybase databases, you need two gateway SIDs, one for each instance of the gateway. If you have only one Sybase database and want to access it sometimes with one set of gateway parameter settings, and other times with different gateway parameter settings, then you will need multiple gateway SIDs for the single Sybase database.

Customize the Initialization Parameter File

The initialization parameter file must be available when the gateway is started. During installation, the following default initialization parameter file is created:

```
$ORACLE_HOME/dg4sybs/admin/initdg4sybs.ora
```

Where \$ORACLE_HOME is the directory under which the gateway is installed.

This initialization file is for the default gateway SID. If you are not using dg4sybs as the gateway SID, you must rename the initialization parameter file using the SID you chose in the preceding step "[Choose a System Identifier for the Gateway](#)" on page 3-1. This default initialization parameter file is sufficient for starting the gateway, verifying a successful installation, and running the demonstration scripts.

A number of initialization parameters can be used to modify the gateway behavior. Refer to [Appendix C, "Initialization Parameters"](#) for the complete list of initialization parameters that can be set. Changes made to the initialization parameters only take effect in the next gateway session. The most important parameter is the HS_FDS_CONNECT_INFO which describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. The syntax for HS_FDS_CONNECT_INFO is as follows:

```
HS_FDS_CONNECT_INFO=host_name:port_number/database_name
```

Where:

Variable	Description
<i>host_name</i>	is the host name or IP address of the machine hosting the Sybase database.
<i>port_number</i>	is the port number of the Sybase database server.
<i>database_name</i>	is the Sybase database name.

See Also: [Appendix C, "Initialization Parameters"](#) and the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about customizing the initialization parameter file.

Configure Oracle Net for the Gateway

The gateway requires Oracle Net to communicate with the Oracle database. After configuring the gateway, perform the following tasks to configure Oracle Net to work with the gateway:

1. [Configure Oracle Net Listener for the Gateway](#)
2. [Stop and Start the Oracle Net Listener for the Gateway](#)

Configure Oracle Net Listener for the Gateway

The Oracle Net Listener listens for incoming requests from the Oracle database. For the Oracle Net Listener to listen for the gateway, information about the gateway must be added to the Oracle Net Listener configuration file, `listener.ora`. This file by default is located in \$ORACLE_HOME/network/admin, where \$ORACLE_HOME is the directory under which the gateway is installed.

The following entries must be added to the `listener.ora` file:

- A list of Oracle Net addresses on which the Oracle Net Listener listens
- The executable name of the gateway that the Oracle Net Listener starts in response to incoming connection requests

A sample of the `listener.ora` entry (`listener.ora.sample`) is available in the `$ORACLE_HOME/dg4sybs/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Syntax of listener.ora File Entries

The Oracle database communicates with the gateway using Oracle Net and any supported protocol adapters. The following is the syntax of the address on which the Oracle Net Listener listens using the TCP/IP protocol adapter:

```
LISTENER=
  (ADDRESS=
    (PROTOCOL=TCP)
    (HOST=host_name)
    (PORT=port_number))
```

Where:

Variable	Description
<code>host_name</code>	is the name of the machine on which the gateway is installed.
<code>port_number</code>	specifies the port number used by the Oracle Net Listener. If you have other listeners running on the same machine, then the value of <code>port_number</code> must be different from the other listeners' port numbers.

To direct the Oracle Net Listener to start the gateway in response to incoming connection requests, add an entry to the `listener.ora` file. The syntax for HP-UX PA-RISC slightly different than the other platforms.

Note: You must use the same SID value in the `listener.ora` file and the `tnsnames.ora` file which will be configured in the next step.

For AIX, Solaris SPARC, and Linux:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4sybs)
    )
  )
```

For HP-UX Itanium:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (ENVS=LD_LIBRARY_PATH=oracle_home_directory/dg4sybs/driver/lib)
      (PROGRAM=dg4sybs)
    )
  )
```

)

For HP-UX PA-RISC:

```
SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(PROGRAM=dg4sybs)
(ENVS=SHLIB_PATH=$ORACLE_HOME/lib32)
)
)
)
```

Where:

Variable	Description
<i>gateway_sid</i>	specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <i>tnsnames.ora</i> file.
<i>oracle_home_directory</i>	specifies the Oracle home directory where the gateway resides.
<i>dg4sybs</i>	specifies the executable name of the Oracle Database Gateway for Sybase.

If you already have an existing Oracle Net Listener, then add the following syntax to *SID_LIST* in the existing *listener.ora* file:

For AIX, Solaris SPARC, and Linux:

```
SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=.
.
)
(SID_DESC=.
.
)
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(PROGRAM=dg4sybs)
)
)
)
```

For HP-UX Itanium:

```
SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=.
.
)
(SID_DESC=.
.
)
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(ENVS=LD_LIBRARY_PATH=oracle_home_directory/dg4sybs/driver/lib)
(PROGRAM=dg4sybs)
)
)
```



```

)
For HP-UX PA-RISC:
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=.
    .
  )
  (SID_DESC=.
    .
  )
  (SID_DESC=
    (SID_NAME=gateway_sid)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4sybs)
    (ENVS=SHLIB_PATH=$ORACLE_HOME/lib32)
  )
)

```

See Also: *Oracle Net Services Administrator's Guide* for information about changing the `listener.ora` file.

Stop and Start the Oracle Net Listener for the Gateway

You must stop and restart the Oracle Net Listener to initiate the new settings, as follows:

1. Set the `PATH` environment variable to `$ORACLE_HOME/bin` where `$ORACLE_HOME` is the directory in which the gateway is installed.

For example on the Linux platform, if you have the Bourne or Korn Shell, enter the following:

```

$ PATH=$ORACLE_HOME/bin:$PATH;export PATH
$ LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH; export LD_LIBRARY_PATH

```

If you have the C Shell, enter the following:

```

$ setenv PATH $ORACLE_HOME/bin:$PATH
$ setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH

```

Table 3–1 specifies which parameter value to use for the different platforms:

Table 3–1 Parameter Values for UNIX Based Platforms

Platform	Parameter Value
Solaris (SPARC) 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
HP-UX PA-RISC	<code>SHLIB_PATH=\$ORACLE_HOME/lib</code>
HP-UX Itanium	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
Linux x86, and Linux x86 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
AIX	<code>LIBPATH=\$ORACLE_HOME/lib</code>

2. If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```

$ lsnrctl stop
$ lsnrctl start

```

3. Check the status of the listener with the new settings, as follows:

```
$ lsnrctl status
```

The following is a partial output from a `lsnrctl` status check:

```
.
.
.
Services Summary...
Service "dg4sybs" has 1 instance(s).
  Instance "dg4sybs", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

In this example, the service name is `dg4sybs` which is the default SID value assigned during installation.

Configure the Oracle Database for Gateway Access

Before you use the gateway to access Sybase data you must configure the Oracle database to enable communication with the gateway over Oracle Net.

To configure the Oracle database you must add connect descriptors to the `tnsnames.ora` file. By default, this file is in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory in which the Oracle database is installed. You cannot use the Oracle Net Assistant or the Oracle Net Easy Config tools to configure the `tnsnames.ora` file. You must edit the file manually.

A sample of the `tnsnames.ora` entry (`tnsnames.ora.sample`) is available in the `$ORACLE_HOME/dg4sybs/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Configuring `tnsnames.ora`

Edit the `tnsnames.ora` file to add a connect descriptor for the gateway. The following is a syntax of the Oracle Net entry using the TCP/IP protocol:

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name)
      (PORT=port_number)
    )
    (CONNECT_DATA=
      (SID=gateway_sid)
    )
    (HS=OK)
  )
```

Where:

Variable	Description
<code>connect_descriptor</code>	is the description of the object to connect to as specified when creating the database link, such as <code>dg4sybs</code> . Check the <code>sqlnet.ora</code> file for the following parameter setting: <ul style="list-style-type: none"> names.directory_path = (TNSNAMES) Note: The <code>sqlnet.ora</code> file is typically stored in <code>\$ORACLE_HOME/network/admin</code> .
TCP	is the TCP protocol used for TCP/IP connections.

Variable	Description
<i>host_name</i>	specifies the machine where the gateway is running.
<i>port_number</i>	matches the port number used by the Oracle Net Listener that is listening for the gateway. The Oracle Net Listener's port number can be found in the <code>listener.ora</code> file used by the Oracle Net Listener. See " Syntax of listener.ora File Entries " on page 3-3.
<i>gateway_sid</i>	specifies the SID of the gateway and matches the SID specified in the <code>listener.ora</code> file of the Oracle Net Listener that is listening for the gateway. See " Configure Oracle Net Listener for the Gateway " on page 3-2 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

See Also: *Oracle Database Administrator's Guide* for information about editing the `tnsnames.ora` file.

Create Database Links

Any Oracle client connected to the Oracle database can access Sybase data through the gateway. The Oracle client and the Oracle database can reside on different machines. The gateway accepts connections only from the Oracle database.

A connection to the gateway is established through a database link when it is first used in an Oracle session. In this context, a connection refers to the connection between the Oracle database and the gateway. The connection remains established until the Oracle session ends. Another session or user can access the same database link and get a distinct connection to the gateway and Sybase database.

Database links are active for the duration of a gateway session. If you want to close a database link during a session, you can do so with the `ALTER SESSION` statement.

To access the Sybase server, you must create a database link. A public database link is the most common of database links.

```
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO
2 "user" IDENTIFIED BY "password" USING 'tns_name_entry';
```

Where:

Variable	Description
<i>dblink</i>	is the complete database link name.
<i>tns_name_entry</i>	specifies the Oracle Net connect descriptor specified in the <code>tnsnames.ora</code> file that identifies the gateway

After the database link is created you can verify the connection to the Sybase database, as follows:

```
SQL> SELECT * FROM DUAL@dblink;
```

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using database links.

Configure Two-Phase Commit

The gateway supports the following transaction capabilities:

- COMMIT_CONFIRM
- READ_ONLY
- SINGLE_SITE

The transaction model is set using the `HS_TRANSACTION_MODEL` initialization parameter. By default, the gateway runs in `COMMIT_CONFIRM` transaction mode. When the Sybase database is updated by a transaction, the gateway becomes the commit point site. The Oracle database commits the unit of work in the Sybase database after verifying that all Oracle databases in the transaction have successfully prepared the transaction. Only one gateway instance can participate in an Oracle two-phase commit transaction as the commit point site.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for information about the two-phase commit process.

To enable the `COMMIT_CONFIRM` transaction mode, perform the following tasks:

1. [Create a Recovery Account and Password](#)
2. [Create the Transaction Log Table](#)

The log table, called `HS_TRANSACTION_LOG`, is where two-phase commit transactions are recorded.

Create a Recovery Account and Password

For the gateway to recover distributed transactions, a recovery account and password must be set up in the Sybase database. By default, both the user name of the account and the password are `RECOVER`. The name of the account can be changed with the gateway initialization parameter `HS_FDS_RECOVERY_ACCOUNT`. The account password can be changed with the gateway initialization parameter `HS_FDS_RECOVERY_PWD`.

Note: Oracle recommends that you do not use the default value `RECOVER` for the user name and password. Moreover, storing plain-text as user name and password in the initialization file is not a good security policy. There is now a utility called `dg4pwd`, that should be used for encryption. Refer to Section 4.2.3, 'Encrypting Initialization parameters' in the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for further details.

1. Set up a user account in the Sybase database. Both the user name and password must be a valid Sybase user name and password.
2. In the initialization parameter file, set the following gateway initialization parameters:
 - `HS_FDS_RECOVERY_ACCOUNT` to the user name of the Sybase user account you set up for recovery.
 - `HS_FDS_RECOVERY_PWD` to the password of the Sybase user account you set up for recovery.

See Also: ["Customize the Initialization Parameter File"](#) on page 3-2 for information about editing the initialization parameter file. For information about HS_FDS_RECOVERY_ACCOUNT and HS_FDS_RECOVERY_PWD, see [Appendix C, "Initialization Parameters"](#).

Create the Transaction Log Table

When configuring the gateway for two-phase commit, a table must be created in the Sybase database for logging transactions. The gateway uses the transaction log table to check the status of failed transactions that were started at the Sybase database by the gateway and registered in the table.

Note: Updates to the transaction log table cannot be part of an Oracle distributed transaction.

Note: The information in the transaction log table is required by the recovery process and must not be altered. The table must be used, accessed, or updated only by the gateway.

The table, called HS_TRANSACTION_LOG, consists of two columns, GLOBAL_TRAN_ID, data type CHAR(64) NOT NULL and TRAN_COMMENT, data type CHAR(255).

You can use another name for the log table, other than HS_TRANSACTION_LOG, by specifying the other name using the HS_FDS_TRANSACTION_LOG initialization parameter.

See Also: [Appendix C, "Initialization Parameters"](#) for information about the HS_FDS_TRANSACTION_LOG initialization parameter.

Create the transaction log table in the user account you created in ["Create a Recovery Account and Password"](#) on page 3-8. Because the transaction log table is used to record the status of a gateway transaction, the table must reside at the database where the Sybase update takes place. Also, the transaction log table must be created under the owner of the recovery account.

Note: To utilize the transaction log table, users of the gateway must be granted privileges on the table.

To create a transaction log table use the dg4sybs_tx.sql script, located in the directory \$ORACLE_HOME/dg4sybs/admin where \$ORACLE_HOME is the directory under which the gateway is installed. Use isql to execute the script, as follows:

```
$ isql -Urecovery_account -Precovery_account_password [-Sserver] -idg4sybs_tx.sql
```

Create Sybase Views for Data Dictionary Support

To enable Oracle data dictionary translation support use the dg4sybs_cvw.sql script, located in the directory \$ORACLE_HOME/dg4sybs/admin where \$ORACLE_HOME is the directory under which the gateway is installed. You must run this script on each Sybase database that you want to access through the gateway. Use isql to execute the script, as follows:

```
$ isql -Usa_user -Psa_pwd [-Sserver] [-Ddatabase] -e -i dg4sybs_cvw.sql
```

where `sa_user` and `sa_pwd` are the Sybase system administrator user ID and password respectively.

Encrypt Gateway Initialization Parameter Values

The gateway uses user IDs and passwords to access the information in the remote database. Some user IDs and passwords must be defined in the gateway initialization file to handle functions such as resource recovery. In the current security conscious environment, having plain-text passwords that are accessible in the initialization file is deemed insecure. The `dg4pwd` encryption utility has been added as part of Heterogeneous Services to help make this more secure. This utility is accessible by this gateway. The initialization parameters which contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using this utility.

Configure the Gateway to Access Multiple Sybase Databases

The tasks for configuring the gateway to access multiple Sybase databases are similar to the tasks for configuring the gateway for a single database. The configuration example assumes the following:

- The gateway is installed and configured with the default SID of `dg4sybs`
- The `ORACLE_HOME` environment variable is set to the directory where the gateway is installed
- The gateway is configured for one Sybase database named `db1`
- Two Sybase databases named `db2` and `db3` on a host with IP Address 204.179.79.15 are being added

Multiple Sybase Databases Example: Configuring the Gateway

Choose One System ID for Each Sybase Database

A separate instance of the gateway is needed for each Sybase database. Each instance needs its own gateway System ID (SID). For this example, the gateway SIDs are chosen for the instances that access the Sybase databases:

- `dg4sybs2` for the gateway accessing database `db2`
- `dg4sybs3` for the gateway accessing database `db3`

Create Two Initialization Parameter Files

Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file, `$ORACLE_HOME/dg4sybs/admin/initdg4sybs.ora`, twice, naming one with the gateway SID for `db2` and the other with the gateway SID for `db3`:

```
$ cd $ORACLE_HOME/dg4sybs/admin
$ cp initdg4sybs.ora initdg4sybs2.ora
$ cp initdg4sybs.ora initdg4sybs3.ora
```

Change the value of the `HS_FDS_CONNECT_INFO` parameter in the new files.

For `initdg4sybs2.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:5000/db2
```

For `initdg4sybs3.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:5000/db3
```

Note: If you have multiple gateway SIDs for the same Sybase database because you want to use different gateway parameter settings at different times, follow the same procedure. You create several initialization parameter files, each with different SIDs and different parameter settings.

Multiple Sybase Databases Example: Configuring Oracle Net Listener

Add Entries to listener.ora

Add two new entries to the Oracle Net Listener configuration file, `listener.ora`. You must have an entry for each gateway instance, even when multiple gateway instances access the same database.

The following example shows the entry for the original installed gateway first, followed by the new entries.

```
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4sybs)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4sybs)
  )
  (SID_DESC=
    (SID_NAME=dg4sybs2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4sybs)
  )
  (SID_DESC=
    (SID_NAME=dg4sybs3)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4sybs)
  )
)
```

where, `oracle_home_directory` is the directory where the gateway resides.

Note: For HP-UX PA-RISC, the `envs` parameter also needs to be set. Refer to "[Syntax of listener.ora File Entries](#)" on page 3-3 for more information about adding the `envs` parameter.

Multiple Sybase Databases Example: Stopping and Starting the Oracle Net Listener

If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

Multiple Sybase Databases Example: Configuring Oracle Database for Gateway Access

Configuring Oracle Net for Multiple Gateway Instances

Add two connect descriptor entries to the `tnsnames.ora` file. You must have an entry for each gateway instance, even if the gateway instances access the same database.

The following Sybase example shows the entry for the original installed gateway first, followed by the two entries for the new gateway instances:

```
old_db_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4sybs) )
  (HS=OK) )
new_db2_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4sybs2) )
  (HS=OK) )
new_db3_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4sybs3) )
  (HS=OK) )
```

The value for `PORT` is the TCP/IP port number of the Oracle Net Listener that is listening for the gateway. The number can be found in the `listener.ora` file used by the Oracle Net Listener. The value for `HOST` is the name of the machine on which the gateway is running. The name also can be found in the `listener.ora` file used by the Oracle Net Listener.

Multiple Sybase Databases Example: Accessing Sybase Data

Enter the following to create a database link for the `dg4sybs2` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK SYBS2 CONNECT TO
  2 "user2" IDENTIFIED BY "password2" USING 'new_db2_using';
```

Enter the following to create a database link for the `dg4sybs3` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK SYBS3 CONNECT TO
  2 "user3" IDENTIFIED BY "password3" USING 'new_db3_using';
```

After the database links are created you can verify the connection to the new Sybase databases, as in the following:

```
SQL> SELECT * FROM ALL_USERS@SYBS2;
```



```
SQL> SELECT * FROM ALL_USERS@SYBS3;
```


Part III

Installing and Configuring Oracle Database Gateway for Informix

Part III, "Installing and Configuring Oracle Database Gateway for Informix" describes how to install and configure Oracle Database Gateway for Informix on UNIX based platforms.

It contains the following chapters:

- Chapter 4, "Installing Oracle Database Gateway for Informix"
- Chapter 5, "Configuring Oracle Database Gateway for Informix"

Installing Oracle Database Gateway for Informix

This chapter provides information about the hardware and software requirements and the installation procedure for Oracle Database Gateway for Informix.

To install the gateway, follow these steps:

1. Ensure that the system meets all of the hardware and software requirements specified in "[System Requirements for Oracle Database Gateway for Informix](#)" on page 4-1.

2. Run the Oracle Universal Installer.

See "[Step Through the Oracle Universal Installer](#)" on page 4-5 for more information about running the Oracle Universal Installer

Oracle Universal Installer is a menu-driven utility that guides you through the installation of the gateway by prompting you with action items. The action items and the sequence in which they appear depend on your platform.

See [Table 4-3](#) for a description of the installation procedure of Oracle Database Gateway for Informix.

System Requirements for Oracle Database Gateway for Informix

This section provides information about the hardware and software requirements for the gateway. It contains the following sections:

- "[Hardware Requirements](#)" on page 4-1
- "[Software Requirements](#)" on page 4-3

Hardware Requirements

[Table 4-1](#) shows the minimum hardware requirements for Oracle Database Gateway for Informix.

Table 4-1 Hardware requirements for Oracle Database Gateway for Informix

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for HP-UX Itanium	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Temporary Disk Space	400 MB	400 MB	400 MB	400 MB	400 MB	400 MB

Table 4–1 (Cont.) Hardware requirements for Oracle Database Gateway for Informix

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for HP-UX Itanium	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Disk Space	1.5 GB	1.5 GB	1.5 GB	750 MB	750 MB	750 MB
Physical Memory*	512 MB	512 MB	512 MB	512 MB	512 MB	512 MB
Swap Space	1 GB	1 GB	1 GB	1 GB	1 GB	1 GB
Processor	IBM RS/6000 AIX-Based System Processor	HP 9000 Series 700 or 800 processor for hp-ux 11.0	HP Itanium processor for hp-ux 11	Sun Solaris Operating System (SPARC) Processor	x86	x86_64

* The minimum swap space is 1 GB (or twice the size of RAM). On systems with 2 GB or more of RAM, the swap space can be between one and two times the size of RAM. On AIX systems with 1 GB or more of memory, do not increase the swap space more than 2 GB.

Checking the Hardware Requirements

To ensure that the system meets the minimum requirements, follow these steps:

1. To determine the physical RAM size, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lssattr -E -l sys0 -a realmem
HP-UX PA-RISC	# /usr/sbin/dmesg grep "Physical:"
HP-UX Itanium	# /usr/contrib/bin/machinfo grep -i Memory
Solaris (SPARC)	# /usr/sbin/prtconf grep "Memory size"
Linux x86	# grep MemTotal /proc/meminfo
Linux x86 64 bit	# grep MemTotal /proc/meminfo

If the size of the physical RAM installed in the system is less than the required size, you must install more memory before continuing.

2. To determine the size of the configured swap space, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lssps -a
HP-UX PA-RISC	# /usr/sbin/swapinfo -a
HP-UX Itanium	# /usr/sbin/swapinfo -a
Solaris (SPARC)	# /usr/sbin/swap -s
Linux x86	# grep SwapTotal /proc/meminfo
Linux x86 64 bit	# grep SwapTotal /proc/meminfo

If necessary, see your operating system documentation for information about how to configure additional swap space.

3. To determine the amount of disk space available in the `/tmp` directory enter the following commands:

Operating System	Command
AIX	# <code>df -k /tmp</code>
HP-UX PA-RISC	# <code>df -k /tmp</code>
HP-UX Itanium	# <code>bdf /tmp</code>
Solaris (SPARC)	# <code>df -k /tmp</code>
Linux x86	# <code>df -k /tmp</code>
Linux x86 64 bit	# <code>df -k /tmp</code>

4. To determine the amount of disk space available on the system enter the following commands:

Operating System	Command
AIX	# <code>df -k</code>
HP-UX PA-RISC	# <code>df -k</code>
HP-UX Itanium	# <code>bdf</code>
Solaris (SPARC)	# <code>df -k</code>
Linux x86	# <code>df -k</code>
Linux x86 64 bit	# <code>df -k</code>

Software Requirements

The following section describes the minimum software requirements for Oracle Database Gateway for Informix.

Operating System

Table 4–2 shows the minimum operating system version required for Oracle Database Gateway for Informix. If your operating system is lower than the minimum requirements, upgrade your operating system to meet the specified levels.

Table 4–2 Operating Systems version for Oracle Database Gateway for Informix

Operating System	Version
AIX	AIX 5L version 5.3, Maintenance level 02 or higher
HP-UX PA-RISC	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
HP-UX Itanium	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
Solaris (SPARC)	Solaris 9 Update 6 or higher or Solaris 10
Linux x86 Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 Suse	SUSE Linux Enterprise Server 10.0

Table 4–2 (Cont.) Operating Systems version for Oracle Database Gateway for Informix

Operating System	Version
Linux x86 64 bit Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 64 bit Suse	SUSE Linux Enterprise Server 10.0
Oracle Enterprise Linux x86	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0
Oracle Enterprise Linux x86 64 bit	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0

Checking the Software Requirements

To ensure that the system meets the minimum requirements, follow these steps:

- To determine which version of AIX is installed, enter the following command:

```
# oslevel -r
```
- To determine which version of HP-UX PA-RISC is installed, enter the following command:

```
# uname -a
```
- To determine which version of HP-UX Itanium is installed, enter the following command:

```
# uname -a
```
- To determine which version of Solaris Operating System (SPARC) is installed, enter the following command:

```
# uname -r
```
- To determine which distribution and version of Linux x86 is installed, enter the following command:

```
# cat /etc/issue
```
- To determine which distribution and version of Linux x86 64 bit is installed, enter the following command:

```
# cat /proc/version
```

Certified Configuration

The gateway supports Informix Dynamic Server. For the latest versions supported refer to the OTN Web site:

<http://www.oracle.com/technology/products/gateways/pdf/certmatrix10g.pdf>

Step Through the Oracle Universal Installer

Table 4–3 describes the installation procedure for Oracle Database Gateway for Informix.

Table 4–3 The Oracle Universal Installer: Steps for Installing the Gateway

Screen	Response
Oracle Universal Installer: Welcome	Click Next .
Oracle Universal Installer: File Locations	<p>The Source section of the screen is where you specify the source location that the Oracle Universal Installer must use to install the Oracle Database Gateway for Informix. You need not edit the file specification in the Path field. The default setting for this field points to the installer file on your Oracle Database Gateway installation media.</p> <p>The Path field in the Destination section of the File Locations screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to <code>ORACLE_HOME</code>. After you set the fields in the File Locations screen as necessary, click Next to continue. After loading the necessary information from the installation media, the Oracle Universal Installer displays the Available Products screen.</p>
Oracle Universal Installer: Available Product Components	<p>a. Select Oracle Database Gateway for Informix 11.1.0.6.0.</p> <p>b. Click Next.</p>
Oracle Database Gateway for Informix	<p>Informix Database Server Host Name - Specify the host name of the machine hosting the Informix database server.</p> <p>Informix Database Server Port number - Specify the port number of the Informix database server</p> <p>Informix Server Name - Specify the Informix server name</p> <p>Informix Database Name - Specify the Informix database name</p> <p>Click Next to continue.</p>
Oracle Universal Installer: Summary	The Installation Summary screen enables you to review a tree list of options and components for this installation. Click Install to start installation.
Oracle Net Configuration Assistant: Welcome	Click Cancel
Oracle Net Configuration Assistant:	Click Yes
Oracle Universal Installer: Configuration Tools	Click Exit
Exit	The final screen of the Oracle Universal Installer is the End of Installation screen. Click Exit to exit the installer.

Configuring Oracle Database Gateway for Informix

After installing the gateway, perform the following tasks to configure Oracle Database Gateway for Informix:

1. [Configure the Gateway Initialization Parameter File](#)
2. [Configure Oracle Net for the Gateway](#)
3. [Configure the Oracle Database for Gateway Access](#)
4. [Create Database Links](#)
5. [Configure Two-Phase Commit](#)
6. [Encrypt Gateway Initialization Parameter Values](#)
7. [Configure the Gateway to Access Multiple Informix Databases](#)

Configure the Gateway Initialization Parameter File

Perform the following tasks to configure the gateway initialization parameter file:

1. [Choose a System Identifier for the Gateway](#)
2. [Customize the Initialization Parameter File](#)

Choose a System Identifier for the Gateway

The gateway system identifier (SID) is an alphanumeric character string that identifies a gateway instance. You need one gateway instance, and therefore one gateway SID, for each Informix database you are accessing. The SID is used as part of the file name for the initialization parameter file. The default SID is `dg4i fmx`.

You can define a gateway SID, but using the default of `dg4i fmx` is easier because you do not need to change the initialization parameter file name. However, if you want to access two Informix databases, you need two gateway SIDs, one for each instance of the gateway. If you have only one Informix database and want to access it sometimes with one set of gateway parameter settings, and other times with different gateway parameter settings, then you will need multiple gateway SIDs for the single Informix database.

Customize the Initialization Parameter File

The initialization parameter file must be available when the gateway is started. During installation, the following default initialization parameter file is created:

`ORACLE_HOME/dg4ifmx/admin/initdg4ifmx.ora`

Where `ORACLE_HOME` is the directory under which the gateway is installed.

This initialization file is for the default gateway SID. If you are not using `dg4ifmx` as the gateway SID, you must rename the initialization parameter file using the SID you chose in the preceding step "[Choose a System Identifier for the Gateway](#)" on page 5-1. This default initialization parameter file is sufficient for starting the gateway, verifying a successful installation, and running the demonstration scripts.

A number of initialization parameters can be used to modify the gateway behavior. Refer to [Appendix C, "Initialization Parameters"](#) for the complete list of initialization parameters that can be set. Changes made to the initialization parameters only take effect in the next gateway session. The most important parameter is the `HS_FDS_CONNECT_INFO` which describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. The syntax for `HS_FDS_CONNECT_INFO` is as follows:

`HS_FDS_CONNECT_INFO=host_name:port_number/server_name/database_name`

Where:

Variable	Description
<code>host_name</code>	is the host name or IP address of the machine hosting the Informix database.
<code>port_number</code>	is the port number of the Informix database server.
<code>server_name</code>	specify the Informix database server name.
<code>database_name</code>	is the Informix database name.

See Also: [Appendix C, "Initialization Parameters"](#) and the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about customizing the initialization parameter file.

Configure Oracle Net for the Gateway

The gateway requires Oracle Net to communicate with the Oracle database. After configuring the gateway, perform the following tasks to configure Oracle Net to work with the gateway:

1. [Configure Oracle Net Listener for the Gateway](#)
2. [Stop and Start the Oracle Net Listener for the Gateway](#)

Configure Oracle Net Listener for the Gateway

The Oracle Net Listener listens for incoming requests from the Oracle database. For the Oracle Net Listener to listen for the gateway, information about the gateway must be added to the Oracle Net Listener configuration file, `listener.ora`. This file by default is located in `ORACLE_HOME/network/admin`, where `ORACLE_HOME` is the directory under which the gateway is installed.

The following entries must be added to the `listener.ora` file:

- A list of Oracle Net addresses on which the Oracle Net Listener listens

- The executable name of the gateway that the Oracle Net Listener starts in response to incoming connection requests

A sample of the `listener.ora` entry (`listener.ora.sample`) is available in the `$ORACLE_HOME/dg4ifmx/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Syntax of listener.ora File Entries

The Oracle database communicates with the gateway using Oracle Net and any supported protocol adapters. The following is the syntax of the address on which the Oracle Net Listener listens using the TCP/IP protocol adapter:

```
LISTENER=
  (ADDRESS=
    (PROTOCOL=TCP)
    (HOST=host_name)
    (PORT=port_number))
```

Where:

Variable	Description
<i>host_name</i>	is the name of the machine on which the gateway is installed.
<i>port_number</i>	specifies the port number used by the Oracle Net Listener. If you have other listeners running on the same machine, the value of <i>port_number</i> must be different from the other listeners' port numbers.

To direct the Oracle Net Listener to start the gateway in response to incoming connection requests, add an entry to the `listener.ora` file. The syntax for HP-UX PA-RISC slightly different than the other platforms.

Note: You must use the same SID value in the `listener.ora` file and the `tnsnames.ora` file which will be configured in the next step.

For AIX, Solaris SPARC, and Linux:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4ifmx)
    )
  )
```

For HP-UX Itanium:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (ENV=LD_LIBRARY_PATH=oracle_home_directory/dg4ifmx/driver/lib)
      (PROGRAM=dg4ifmx)
    )
  )
```

For HP-UX PA-RISC:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4ifmx)
      (ENVS=SHLIB_PATH=oracle_home_directory/lib32)
    )
  )
)
```

Where:

Variable	Description
<i>gateway_sid</i>	specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <code>tnsnames.ora</code> file.
<i>oracle_home_directory</i>	specifies the Oracle home directory where the gateway resides.
<i>dg4ifmx</i>	specifies the executable name of the Oracle Database Gateway for Informix.

If you already have an existing Oracle Net Listener, then add the following syntax to `SID_LIST` in the existing `listener.ora` file:

For AIX, Solaris SPARC, and Linux:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=.
      .
    )
    (SID_DESC=.
      .
    )
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4ifmx)
    )
  )
)
```

For HP-UX Itanium:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=.
      .
    )
    (SID_DESC=.
      .
    )
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (ENVS=LD_LIBRARY_PATH=oracle_home_directory/dg4ifmx/driver/lib)
      (PROGRAM=dg4ifmx)
    )
  )
)
```

```

)
For HP-UX PA-RISC:
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=.
    .
  )
  (SID_DESC=.
    .
  )
  (SID_DESC=
    (SID_NAME=gateway_sid)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4ifmx)
    (ENVS=SHLIB_PATH=$ORACLE_HOME/lib32)
  )
)

```

See Also: *Oracle Database Net Services Administrator's Guide* for information about changing the `listener.ora` file.

Stop and Start the Oracle Net Listener for the Gateway

You must stop and restart the Oracle Net Listener to initiate the new settings, as follows:

1. Set the `PATH` environment variable to `$ORACLE_HOME/bin` where `$ORACLE_HOME` is the directory in which the gateway is installed.

For example on the Linux platform, if you have the Bourne or Korn Shell, enter the following:

```

$ PATH=$ORACLE_HOME/bin:$PATH;export PATH
$ LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH; export LD_LIBRARY_PATH

```

If you have the C Shell, enter the following:

```

$ setenv PATH $ORACLE_HOME/bin:$PATH
$ setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH

```

[Table 5–1](#) specifies which parameter value to use for the different platforms:

Table 5–1 Parameter Values for UNIX Based Platforms

Platform	Parameter Value
Solaris (SPARC) 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
HP-UX PA-RISC	<code>SHLIB_PATH=\$ORACLE_HOME/lib</code>
HP-UX Itanium	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
Linux x86, and Linux x86 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
AIX	<code>LIBPATH=\$ORACLE_HOME/lib</code>

2. If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```

$ lsnrctl stop
$ lsnrctl start

```

3. Check the status of the listener with the new settings, as follows:

```
$ lsnrctl status
```

The following is a partial output from a `lsnrctl` status check:

```
.
.
.
Services Summary...
Service "dg4ifmx" has 1 instance(s).
  Instance "dg4ifmx", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

In this example, the service name is `dg4ifmx` which is the default SID value assigned during installation.

Configure the Oracle Database for Gateway Access

Before you use the gateway to access Informix data you must configure the Oracle database to enable communication with the gateway over Oracle Net.

To configure the Oracle database you must add connect descriptors to the `tnsnames.ora` file. By default, this file is in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory in which the Oracle database is installed. You cannot use the Oracle Net Assistant or the Oracle Net Easy Config tools to configure the `tnsnames.ora` file. You must edit the file manually.

A sample of the `tnsnames.ora` entry (`tnsnames.ora.sample`) is available in the `$ORACLE_HOME/dg4ifmx/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Configuring `tnsnames.ora`

Edit the `tnsnames.ora` file to add a connect descriptor for the gateway. The following is a syntax of the Oracle Net entry using the TCP/IP protocol:

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name)
      (PORT=port_number)
    )
    (CONNECT_DATA=
      (SID=gateway_sid)
    )
    (HS=OK)
  )
```

Where:

Variable	Description
<code>connect_descriptor</code>	is the description of the object to connect to as specified when creating the database link, such as <code>dg4ifmx</code> . Check the <code>sqlnet.ora</code> file for the following parameter setting: <ul style="list-style-type: none"> names.directory_path = (TNSNAMES) Note: The <code>sqlnet.ora</code> file is typically stored in <code>\$ORACLE_HOME/network/admin</code> .
TCP	is the TCP protocol used for TCP/IP connections.

Variable	Description
<i>host_name</i>	specifies the machine where the gateway is running.
<i>port_number</i>	matches the port number used by the Oracle Net Listener that is listening for the gateway. The Oracle Net Listener's port number can be found in the <code>listener.ora</code> file used by the Oracle Net Listener. See "Syntax of listener.ora File Entries" on page 5-3.
<i>gateway_sid</i>	specifies the SID of the gateway and matches the SID specified in the <code>listener.ora</code> file of the Oracle Net Listener that is listening for the gateway. See "Configure Oracle Net Listener for the Gateway" on page 5-2 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

See Also: *Oracle Database Administrator's Guide* for information about editing the `tnsnames.ora` file.

Create Database Links

Any Oracle client connected to the Oracle database can access Informix data through the gateway. The Oracle client and the Oracle database can reside on different machines. The gateway accepts connections only from the Oracle database.

A connection to the gateway is established through a database link when it is first used in an Oracle session. In this context, a connection refers to the connection between the Oracle database and the gateway. The connection remains established until the Oracle session ends. Another session or user can access the same database link and get a distinct connection to the gateway and Informix database.

Database links are active for the duration of a gateway session. If you want to close a database link during a session, you can do so with the `ALTER SESSION` statement.

To access the Informix server, you must create a database link. A public database link is the most common of database links.

```
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO
2 "user" IDENTIFIED BY "password" USING 'tns_name_entry';
```

Where:

Variable	Description
<i>dblink</i>	is the complete database link name.
<i>tns_name_entry</i>	specifies the Oracle Net connect descriptor specified in the <code>tnsnames.ora</code> file that identifies the gateway

After the database link is created you can verify the connection to the Informix database, as follows:

```
SQL> SELECT * FROM DUAL@dblink;
```

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using database links.

Configure Two-Phase Commit

The gateway supports the following transaction capabilities:

- COMMIT_CONFIRM
- READ_ONLY
- SINGLE_SITE

The transaction model is set using the HS_TRANSACTION_MODEL initialization parameter. By default, the gateway runs in COMMIT_CONFIRM transaction mode. When the Informix database is updated by a transaction, the gateway becomes the commit point site. The Oracle database commits the unit of work in the Informix database after verifying that all Oracle databases in the transaction have successfully prepared the transaction. Only one gateway instance can participate in an Oracle two-phase commit transaction as the commit point site.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for information about the two-phase commit process.

To enable the COMMIT_CONFIRM transaction mode, perform the following tasks:

1. [Create a Recovery Account and Password](#)
2. [Create the Transaction Log Table](#)

The log table, called HS_TRANSACTION_LOG, is where two-phase commit transactions are recorded.

Create a Recovery Account and Password

For the gateway to recover distributed transactions, a recovery account and password must be set up in the Informix database. By default, both the user name of the account and the password are RECOVER. The name of the account can be changed with the gateway initialization parameter HS_FDS_RECOVERY_ACCOUNT. The account password can be changed with the gateway initialization parameter HS_FDS_RECOVERY_PWD.

Note: Oracle recommends that you do not use the default value RECOVER for the user name and password. Moreover, storing plain-text as user name and password in the initialization file is not a good security policy. There is now a utility called dg4pwd, that should be used for encryption. Refer to Section 4.2.3, 'Encrypting Initialization parameters' in the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for further details.

1. Set up a user account in the Informix database. Both the user name and password must be a valid Informix user name and password.
2. In the initialization parameter file, set the following gateway initialization parameters:
 - HS_FDS_RECOVERY_ACCOUNT to the user name of the Informix user account you set up for recovery.
 - HS_FDS_RECOVERY_PWD to the password of the Informix user account you set up for recovery.

See Also: [Customize the Initialization Parameter File](#) on page 5-1 for information about editing the initialization parameter file. For information about HS_FDS_RECOVERY_ACCOUNT and HS_FDS_RECOVERY_PWD, see [Appendix C, "Initialization Parameters"](#).

Create the Transaction Log Table

When configuring the gateway for two-phase commit, a table must be created in the Informix database for logging transactions. The gateway uses the transaction log table to check the status of failed transactions that were started at the Informix database by the gateway and registered in the table.

Note: Updates to the transaction log table cannot be part of an Oracle distributed transaction.

Note: The information in the transaction log table is required by the recovery process and must not be altered. The table must be used, accessed, or updated only by the gateway.

The table, called HS_TRANSACTION_LOG, consists of two columns, GLOBAL_TRAN_ID, data type CHAR(64) NOT NULL and TRAN_COMMENT, data type CHAR(255).

You can use another name for the log table, other than HS_TRANSACTION_LOG, by specifying the other name using the HS_FDS_TRANSACTION_LOG initialization parameter.

See Also: [Appendix C, "Initialization Parameters"](#) for information about the HS_FDS_TRANSACTION_LOG initialization parameter.

Create the transaction log table in the user account you created in [Create a Recovery Account and Password](#) on page 5-8. Because the transaction log table is used to record the status of a gateway transaction, the table must reside at the database where the Informix update takes place. Also, the transaction log table must be created under the owner of the recovery account.

Note: To utilize the transaction log table, users of the gateway must be granted privileges on the table.

To create a transaction log table use the dg4ifmx_tx.sql script, located in the directory \$ORACLE_HOME/dg4ifmx/admin where \$ORACLE_HOME is the directory under which the gateway is installed, as follows:

1. Login as user ID RECOVER.
2. Set environment variable DELIMIDENT.

If you have the Bourne or Korn Shell, enter the following:

```
$ DELIMIDENT = y; export DELIMIDENT
```

If you have the C Shell, enter the following:

```
$ setenv DELIMIDENT y
```

3. Execute the script using dbaccess, as follows.

```
$ cd $ORACLE_HOME/dg4ifmx/admin
$ dbaccess database_name dg4ifmx_tx.sql
```

Encrypt Gateway Initialization Parameter Values

The gateway uses user IDs and passwords to access the information in the remote database. Some user IDs and passwords must be defined in the gateway initialization file to handle functions such as resource recovery. In the current security conscious environment, having plain-text passwords that are accessible in the initialization file is deemed insecure. The `dg4pwd` encryption utility has been added as part of Heterogeneous Services to help make this more secure. This utility is accessible by this gateway. The initialization parameters which contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using this utility.

Configure the Gateway to Access Multiple Informix Databases

The tasks for configuring the gateway to access multiple Informix databases are similar to the tasks for configuring the gateway for a single database. The configuration example assumes the following:

- The gateway is installed and configured with the default SID of `dg4ifmx`.
- The `ORACLE_HOME` environment variable is set to the directory where the gateway is installed.
- The gateway is configured for one Informix database named `db1`.
- Two Informix databases named `db2` and `db3` on a host with IP Address 204.179.79.15 are being added.

Multiple Informix Databases Example: Configuring the Gateway

Choose One System ID for Each Informix Database

A separate instance of the gateway is needed for each Informix database. Each instance needs its own gateway System ID (SID). For this example, the gateway SIDs are chosen for the instances that access the Informix databases:

- `dg4ifmx2` for the gateway accessing database `db2`.
- `dg4ifmx3` for the gateway accessing database `db3`.

Create Two Initialization Parameter Files

Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file, `$ORACLE_HOME/dg4ifmx/admin/initdg4ifmx.ora`, twice, naming one with the gateway SID for `db2` and the other with the gateway SID for `db3`:

```
$ cd $ORACLE_HOME/dg4ifmx/admin
$ cp initdg4ifmx.ora initdg4ifmx2.ora
$ cp initdg4ifmx.ora initdg4ifmx3.ora
```

Change the value of the `HS_FDS_CONNECT_INFO` parameter in the new files.

For `initdg4ifmx2.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:3900/sr2/db2
```

For `initdg4ifmx3.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:3900/sr3/db3
```

Note: If you have multiple gateway SIDs for the same Informix database because you want to use different gateway parameter settings at different times, follow the same procedure. You create several initialization parameter files, each with different SIDs and different parameter settings.

Multiple Informix Databases Example: Configuring Oracle Net Listener

Add Entries to listener.ora

Add two new entries to the Oracle Net Listener configuration file, `listener.ora`. You must have an entry for each gateway instance, even when multiple gateway instances access the same database.

The following example shows the entry for the original installed gateway first, followed by the new entries.

```
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4ifmx)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4ifmx)
  )
  (SID_DESC=
    (SID_NAME=dg4ifmx2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4ifmx)
  )
  (SID_DESC=
    (SID_NAME=dg4ifmx3)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4ifmx)
  )
)
```

where, `oracle_home_directory` is the directory where the gateway resides.

Note: For HP-UX PA-RISC, the `envs` parameter also needs to be set. Refer to "[Syntax of listener.ora File Entries](#)" on page 5-3 for more information on adding the `envs` parameter.

Multiple Informix Databases Example: Stopping and Starting the Oracle Net Listener

If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

Multiple Informix Databases Example: Configuring Oracle Database for Gateway Access

Configuring Oracle Net for Multiple Gateway Instances

Add two connect descriptor entries to the `tnsnames.ora` file. You must have an entry for each gateway instance, even if the gateway instances access the same database.

The following Informix example shows the entry for the original installed gateway first, followed by the two entries for the new gateway instances:

```
old_db_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name))
  (CONNECT_DATA=
    (SID=dg4ifmx))
  (HS=OK))
new_db2_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name))
  (CONNECT_DATA=
    (SID=dg4ifmx2))
  (HS=OK))
new_db3_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name))
  (CONNECT_DATA=
    (SID=dg4ifmx3))
  (HS=OK))
```

The value for `PORT` is the TCP/IP port number of the Oracle Net Listener that is listening for the gateway. The number can be found in the `listener.ora` file used by the Oracle Net Listener. The value for `HOST` is the name of the machine on which the gateway is running. The name also can be found in the `listener.ora` file used by the Oracle Net Listener.

Multiple Informix Databases Example: Accessing Informix Data

Enter the following to create a database link for the `dg4ifmx2` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK IFMX2 CONNECT TO
  2 "user2" IDENTIFIED BY "password2" USING 'new_db2_using';
```

Enter the following to create a database link for the `dg4ifmx3` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK IFMX3 CONNECT TO
  2 "user3" IDENTIFIED BY "password3" USING 'new_db3_using';
```

After the database links are created you can verify the connection to the new Informix databases, as in the following:

```
SQL> SELECT * FROM ALL_USERS@IFMX2;
```

```
SQL> SELECT * FROM ALL_USERS@IFMX3;
```


Part IV

Installing and Configuring Oracle Database Gateway for Teradata

Part IV, "Installing and Configuring Oracle Database Gateway for Teradata" describes how to install and configure Oracle Database Gateway for Teradata on UNIX based platforms.

It contains the following chapters:

- [Chapter 6, "Installing Oracle Database Gateway for Teradata"](#)
- [Chapter 7, "Configuring Oracle Database Gateway for Teradata"](#)

Installing Oracle Database Gateway for Teradata

This chapter provides information about the hardware and software requirements and the installation procedure for Oracle Database Gateway for Teradata.

To install the gateway, follow these steps:

1. Ensure that the system meets all of the hardware and software requirements specified in "[System Requirements for Oracle Database Gateway for Teradata](#)" on page 6-1.

2. Run the Oracle Universal Installer.

See "[Step Through the Oracle Universal Installer](#)" on page 6-4 for more information about running the Oracle Universal Installer.

Oracle Universal Installer is a menu-driven utility that guides you through the installation of the gateway by prompting you with action items. The action items and the sequence in which they appear depend on your platform.

See [Table 6-3](#) for a description of the installation procedure of Oracle Database Gateway for Teradata

System Requirements for Oracle Database Gateway for Teradata

This section provides information about the hardware and software requirements for the gateway. It contains the following sections:

- "[Hardware Requirements](#)" on page 6-1
- "[Software Requirements](#)" on page 6-3

Hardware Requirements

[Table 6-1](#) shows the minimum hardware requirements for Oracle Database Gateway for Teradata.

Table 6-1 Hardware requirements for Oracle Database Gateway for Teradata

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for HP-UX Itanium	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Temporary Disk Space	400 MB	400 MB	400 MB	400 MB	400 MB	400 MB
Disk Space	1.5 GB	1.5 GB	1.5 GB	750 MB	750 MB	750 MB

Table 6–1 (Cont.) Hardware requirements for Oracle Database Gateway for Teradata

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for HP-UX Itanium	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Physical Memory*	512 MB	512 MB	512 MB	512 MB	512 MB	512 MB
Swap Space	1 GB	1 GB	1 GB	1 GB	1 GB	1 GB
Processor	IBM RS/6000 AIX-Based System Processor	HP 9000 Series 700 or 800 processor for hp-ux 11.0	HP Itanium processor for hp-ux 11	Sun Solaris Operating System (SPARC) Processor	x86	x86_64

* The minimum swap space is 1 GB (or twice the size of RAM). On systems with 2 GB or more of RAM, the swap space can be between one and two times the size of RAM. On AIX systems with 1 GB or more of memory, do not increase the swap space more than 2 GB.

Checking the Hardware Requirements

To ensure that the system meets the minimum requirements, follow these steps:

1. To determine the physical RAM size, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lstat -E -l sys0 -a realmem
HP-UX PA-RISC	# /usr/sbin/dmesg grep "Physical:"
HP-UX Itanium	# /usr/contrib/bin/machinfo grep -i Memory
Solaris (SPARC)	# /usr/sbin/prtconf grep "Memory size"
Linux x86	# grep MemTotal /proc/meminfo
Linux x86 64 bit	# grep MemTotal /proc/meminfo

If the size of the physical RAM installed in the system is less than the required size, you must install more memory before continuing.

2. To determine the size of the configured swap space, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lspcs -a
HP-UX PA-RISC	# /usr/sbin/swapinfo -a
HP-UX Itanium	# /usr/sbin/swapinfo -a
Solaris (SPARC)	# /usr/sbin/swap -s
Linux x86	# grep SwapTotal /proc/meminfo
Linux x86 64 bit	# grep SwapTotal /proc/meminfo

If necessary, see your operating system documentation for information about how to configure additional swap space.

3. To determine the amount of disk space available in the /tmp directory enter the following commands:

Operating System	Command
AIX	# df -k /tmp
HP-UX PA-RISC	# df -k /tmp
HP-UX Itanium	# bdf /tmp
Solaris (SPARC)	# df -k /tmp
Linux x86	# df -k /tmp
Linux x86 64 bit	# df -k /tmp

4. To determine the amount of disk space available on the system enter the following commands:

Operating System	Command
AIX	# df -k
HP-UX PA-RISC	# df -k
HP-UX Itanium	# bdf
Solaris (SPARC)	# df -k
Linux x86	# df -k
Linux x86 64 bit	# df -k

Software Requirements

The following section describes the minimum software requirements for Oracle Database Gateway for Teradata.

Operating System

Table 6–2 shows the minimum operating system version required for Oracle Database Gateway for Teradata. If your operating system is lower than the minimum requirements, upgrade your operating system to meet the specified levels.

Table 6–2 Operating Systems version for Oracle Database Gateway for Teradata

Operating System	Version
AIX	AIX 5L version 5.3, Maintenance level 02 or higher
HP-UX PA-RISC	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
HP-UX Itanium	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
Solaris (SPARC)	Solaris 9 Update 6 or higher or Solaris 10
Linux x86 Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 Suse	SUSE Linux Enterprise Server 10.0
Linux x86 64 bit Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 64 bit Suse	SUSE Linux Enterprise Server 10.0

Table 6–2 (Cont.) Operating Systems version for Oracle Database Gateway for Teradata

Operating System	Version
Oracle Enterprise Linux x86	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0
Oracle Enterprise Linux x86 64 bit	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0

Checking the Software Requirements

To ensure that the system meets the minimum requirements, follow these steps:

- To determine which version of AIX is installed, enter the following command:

```
# oslevel -r
```
- To determine which version of HP-UX PA-RISC is installed, enter the following command:

```
# uname -a
```
- To determine which version of HP-UX Itanium is installed, enter the following command:

```
# uname -a
```
- To determine which version of Solaris Operating System (SPARC) is installed, enter the following command:

```
# uname -r
```
- To determine which distribution and version of Linux x86 is installed, enter the following command:

```
# cat /etc/issue
```
- To determine which distribution and version of Linux x86 64 bit is installed, enter the following command:

```
# cat /proc/version
```

Certified Configuration

Teradata client libraries are required on the machine where the gateway is installed. For the latest certified clients refer to the OTN Web site:

<http://www.oracle.com/technology/products/gateways/pdf/certmatrix10g.pdf>

Step Through the Oracle Universal Installer

Table 6–3 describes the installation procedure for Oracle Database Gateway for Teradata.

Table 6–3 The Oracle Universal Installer: Steps for Installing the Gateway

Screen	Response
Oracle Universal Installer: Welcome	Click Next .
Oracle Universal Installer: File Locations	<p>The Source section of the screen is where you specify the source location that the Oracle Universal Installer must use to install the Oracle Database Gateway for Teradata. You need not edit the file specification in the Path field. The default setting for this field points to the installer file on your Oracle Database Gateway installation media.</p> <p>The Path field in the Destination section of the File Locations screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to <code>ORACLE_HOME</code>. After you set the fields in the File Locations screen as necessary, click Next to continue. After loading the necessary information from the installation media, the Oracle Universal Installer displays the Available Products screen.</p>
Oracle Universal Installer: Available Product Components	<p>a. Select Oracle Database Gateway for Teradata 11.1.0.6.0.</p> <p>b. Click Next.</p>
Oracle Database Gateway for Teradata	<p>Teradata Database Server Host IP or Alias - Specify either the host IP or alias name of the machine running the Teradata database server.</p> <p>Teradata Database Server Port number - Specify the port number of the Teradata database server</p> <p>Teradata Database Name - Specify the Teradata database name</p> <p>Teradata TD_ICU_DATA Path - Specify the local path where ICU data libraries are located (Typically <code>/opt/teradata/tdicu/lib</code> or what <code>\$TD_ICU_DATA</code> is set to in <code>/etc/profile</code>).</p> <p>Teradata COPLIB Path - Specify the local path where COPLIB is located (Typically <code>/usr/lib</code> or what <code>\$COPLIB</code> is set to in <code>/etc/profile</code>).</p> <p>Teradata COPERR Path - Specify the local path where COPERR is located (Typically <code>/usr/lib</code> or what <code>\$COPERR</code> is set to in <code>/etc/profile</code>).</p> <p>Click Next to continue.</p>
Oracle Universal Installer: Summary	The Installation Summary screen enables you to review a tree list of options and components for this installation. Click Install to start installation.
Oracle Net Configuration Assistant: Welcome	Click Cancel
Oracle Net Configuration Assistant:	Click Yes
Oracle Universal Installer: Configuration Tools	Click Exit
Exit	The final screen of the Oracle Universal Installer is the End of Installation screen. Click Exit to exit the installer.

Configuring Oracle Database Gateway for Teradata

After installing the gateway, perform the following tasks to configure Oracle Database Gateway for Teradata:

1. [Configure the Gateway Initialization Parameter File](#)
2. [Configure Oracle Net for the Gateway](#)
3. [Configure the Oracle Database for Gateway Access](#)
4. [Create Database Links](#)
5. [Configure Two-Phase Commit](#)
6. [Encrypt Gateway Initialization Parameter Values](#)
7. [Configure the Gateway to Access Multiple Teradata Databases](#)

Configure the Gateway Initialization Parameter File

Perform the following tasks to configure the gateway initialization parameter file:

1. [Choose a System Identifier for the Gateway](#)
2. [Customize the Initialization Parameter File](#)

Choose a System Identifier for the Gateway

The gateway system identifier (SID) is an alphanumeric character string that identifies a gateway instance. You need one gateway instance, and therefore one gateway SID, for each Teradata database you are accessing. The SID is used as part of the file name for the initialization parameter file. The default SID is `dg4tera`.

You can define a gateway SID, but using the default of `dg4tera` is easier because you do not need to change the initialization parameter file name. However, if you want to access two Teradata databases, you need two gateway SIDs, one for each instance of the gateway. If you have only one Teradata database and want to access it sometimes with one set of gateway parameter settings, and other times with different gateway parameter settings, then you will need multiple gateway SIDs for the single Teradata database.

Customize the Initialization Parameter File

The initialization parameter file must be available when the gateway is started. During installation, the following default initialization parameter file is created:

```
$ORACLE_HOME/dg4tera/admin/initdg4tera.ora
```

Where \$ORACLE_HOME is the directory under which the gateway is installed.

This initialization file is for the default gateway SID. If you are not using dg4tera as the gateway SID, you must rename the initialization parameter file using the SID you chose in the preceding step "[Choose a System Identifier for the Gateway](#)" on page 7-1. This default initialization parameter file is sufficient for starting the gateway, verifying a successful installation, and running the demonstration scripts.

A number of initialization parameters can be used to modify the gateway behavior. Refer to [Appendix C, "Initialization Parameters"](#) for the complete list of initialization parameters that can be set. Changes made to the initialization parameters only take effect in the next gateway session. The most important parameter is the HS_FDS_CONNECT_INFO which describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. The syntax for HS_FDS_CONNECT_INFO is as follows:

```
HS_FDS_CONNECT_INFO=host_alias:port_number[/database_name]
```

Where:

Variable	Description
host_alias	is the host alias name or IP address of the machine hosting the Teradata database.
port_number	is the port number of the Teradata database server.
database_name	is the Teradata database name.

See Also: [Appendix C, "Initialization Parameters"](#) and the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about customizing the initialization parameter file.

Configure Oracle Net for the Gateway

The gateway requires Oracle Net to communicate with the Oracle database. After configuring the gateway, perform the following tasks to configure Oracle Net to work with the gateway:

1. [Configure Oracle Net Listener for the Gateway](#)
2. [Stop and Start the Oracle Net Listener for the Gateway](#)

Configure Oracle Net Listener for the Gateway

The Oracle Net Listener listens for incoming requests from the Oracle database. For the Oracle Net Listener to listen for the gateway, information about the gateway must be added to the Oracle Net Listener configuration file, listener.ora. This file by default is located in \$ORACLE_HOME/network/admin, where \$ORACLE_HOME is the directory under which the gateway is installed.

The following entries must be added to the listener.ora file:

- A list of Oracle Net addresses on which the Oracle Net Listener listens.
- The executable name of the gateway that the Oracle Net Listener starts in response to incoming connection requests.

A sample of the `listener.ora` entry (`listener.ora.sample`) is available in the `$ORACLE_HOME/dg4tera/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Syntax of listener.ora File Entries

The Oracle database communicates with the gateway using Oracle Net and any supported protocol adapters. The following is the syntax of the address on which the Oracle Net Listener listens using the TCP/IP protocol adapter:

```
LISTENER=
  (ADDRESS=
    (PROTOCOL=TCP)
    (HOST=host_name)
    (PORT=port_number))
```

Where:

Variable	Description
<code>host_name</code>	is the name of the machine on which the gateway is installed.
<code>port_number</code>	specifies the port number used by the Oracle Net Listener. If you have other listeners running on the same machine, then the value of <code>port_number</code> must be different from the other listeners' port numbers.

To direct the Oracle Net Listener to start the gateway in response to incoming connection requests, add an entry to the `listener.ora` file.

Note: You must use the same SID value in the `listener.ora` file and the `tnsnames.ora` file which will be configured in the next step.

For Linux 32bit:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4tera)
      (ENVS=LD_LIBRARY_PATH=oracle_home_directory/lib:teradata_client_library_
directory:/usr/lib)
    )
  )
```

For Solaris SPARC and Linux x86 64bit:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4tera)
      (ENVS=LD_LIBRARY_PATH=teradata_client_library_directory:oracle_home_
directory/lib32:/usr/lib)
    )
  )
```

For AIX:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4tera)
      (ENVS=LIBPATH=teradata_client_library_directory:oracle_home_
directory/lib32:/usr/lib)
    )
  )
```

For HP-UX PA-RISC:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4tera)
      (ENVS=SHLIB_PATH=teradata_client_library_directory:oracle_home_
directory/lib32:/usr/lib)
    )
  )
```

For HP-UX Itanium:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4tera)
      (ENVS=LD_LIBRARY_PATH=oracle_home_directory/dg4tera/driver/lib:oracle_
home_directory/lib:teradata_client_library_directory:/usr/lib/hpux64)
    )
  )
```

Where:

Variable	Description
<i>gateway_sid</i>	specifies the SID of the gateway. Matches the gateway SID specified in the connect descriptor entry in the <code>tnsnames.ora</code> file.
<i>oracle_home_directory</i>	specifies the Oracle home directory where the gateway resides.
<i>teradata_client_library_directory</i>	specifies the directory where the Teradata client directory resides.
<i>dg4tera</i>	specifies the executable name of the Oracle Database Gateway for Teradata.

If you already have an existing Oracle Net Listener, then add the following syntax to `SID_LIST` in the existing `listener.ora` file. Note the syntax provided below is for Linux 32 bit. Refer to the above section for other platforms.

For Linux 32 bit:

```
SID_LIST_LISTENER=
```

```
(SID_LIST=
  (SID_DESC=.
    .
  )
  (SID_DESC=.
    .
  )
  (SID_DESC=
    (SID_NAME=gateway_sid)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4tera)
  )
  (ENVS=LD_LIBRARY_PATH=oracle_home_directory/lib:teradata_client_library_
  directory:/usr/lib)
  )
)
```

See Also: *Oracle Net Services Administrator's Guide* for information about changing the `listener.ora` file.

Stop and Start the Oracle Net Listener for the Gateway

You must stop and restart the Oracle Net Listener to initiate the new settings, as follows:

1. Set the `PATH` environment variable to `$ORACLE_HOME/bin` where `$ORACLE_HOME` is the directory in which the gateway is installed.

For example on the Linux platform, if you have the Bourne or Korn Shell, enter the following:

```
$ PATH=$ORACLE_HOME/bin:$PATH;export PATH
$ LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH; export LD_LIBRARY_PATH
```

If you have the C Shell, enter the following:

```
$ setenv PATH $ORACLE_HOME/bin:$PATH
$ setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH
```

[Table 7-1](#) specifies which parameter value to use for the different platforms:

Table 7-1 Parameter Values for UNIX Based Platforms

Platform	Parameter Value
Solaris (SPARC) 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
HP-UX PA-RISC	<code>SHLIB_PATH=\$ORACLE_HOME/lib</code>
HP-UX Itanium	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
Linux x86, and Linux x86 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
AIX	<code>LIBPATH=\$ORACLE_HOME/lib</code>

2. If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

3. Check the status of the listener with the new settings, as follows:

```
$ lsnrctl status
```

The following is a partial output from a `lsnrctl` status check:

```
.
.
.
Services Summary...
Service "dg4tera" has 1 instance(s).
  Instance "dg4tera", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

In this example, the service name is `dg4tera` which is the default SID value assigned during installation.

Configure the Oracle Database for Gateway Access

Before you use the gateway to access Teradata data you must configure the Oracle database to enable communication with the gateway over Oracle Net.

To configure the Oracle database you must add connect descriptors to the `tnsnames.ora` file. By default, this file is in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory in which the Oracle database is installed. You cannot use the Oracle Net Assistant or the Oracle Net Easy Config tools to configure the `tnsnames.ora` file. You must edit the file manually.

A sample of the `tnsnames.ora` entry (`tnsnames.ora.sample`) is available in the `$ORACLE_HOME/dg4tera/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Configuring `tnsnames.ora`

Edit the `tnsnames.ora` file to add a connect descriptor for the gateway. The following is a syntax of the Oracle Net entry using the TCP/IP protocol:

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name)
      (PORT=port_number)
    )
    (CONNECT_DATA=
      (SID=gateway_sid)
    (HS=OK) )
```

Where:

Variable	Description
<code>connect_descriptor</code>	is the description of the object to connect to as specified when creating the database link, such as <code>dg4tera</code> . Check the <code>sqlnet.ora</code> file for the following parameter setting: <code>names.directory_path = (TNSNAMES)</code> Note: The <code>sqlnet.ora</code> file is typically stored in <code>\$ORACLE_HOME/network/admin</code> .
TCP	is the TCP protocol used for TCP/IP connections.
<code>host_name</code>	specifies the machine where the gateway is running.

Variable	Description
<code>port_number</code>	matches the port number used by the Oracle Net Listener that is listening for the gateway. The Oracle Net Listener's port number can be found in the <code>listener.ora</code> file used by the Oracle Net Listener. See "Syntax of <code>listener.ora</code> File Entries" on page 7-3.
<code>gateway_sid</code>	specifies the SID of the gateway and matches the SID specified in the <code>listener.ora</code> file of the Oracle Net Listener that is listening for the gateway. See "Configure Oracle Net Listener for the Gateway" on page 7-2 for more information.
<code>(HS=OK)</code>	specifies that this connect descriptor connects to a non-Oracle system.

See Also: *Oracle Database Administrator's Guide* for information about editing the `tnsnames.ora` file.

Create Database Links

Any Oracle client connected to the Oracle database can access Teradata data through the gateway. The Oracle client and the Oracle database can reside on different machines. The gateway accepts connections only from the Oracle database.

A connection to the gateway is established through a database link when it is first used in an Oracle session. In this context, a connection refers to the connection between the Oracle database and the gateway. The connection remains established until the Oracle session ends. Another session or user can access the same database link and get a distinct connection to the gateway and Teradata database.

Database links are active for the duration of a gateway session. If you want to close a database link during a session, you can do so with the `ALTER SESSION` statement.

To access the Teradata server, you must create a database link. A public database link is the most common of database links.

```
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO
2 "user" IDENTIFIED BY "password" USING 'tns_name_entry';
```

Where:

Variable	Description
<code>dblink</code>	is the complete database link name.
<code>tns_name_entry</code>	specifies the Oracle Net connect descriptor specified in the <code>tnsnames.ora</code> file that identifies the gateway

After the database link is created you can verify the connection to the Teradata database, as follows:

```
SQL> SELECT * FROM DUAL@dblink;
```

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Services Administrator's Guide* for more information about using database links.

Configure Two-Phase Commit

The gateway supports the following transaction capabilities:

- COMMIT_CONFIRM
- READ_ONLY
- SINGLE_SITE

The transaction model is set using the HS_TRANSACTION_MODEL initialization parameter. By default, the gateway runs in COMMIT_CONFIRM transaction mode. When the Teradata database is updated by a transaction, the gateway becomes the commit point site. The Oracle database commits the unit of work in the Teradata database after verifying that all Oracle databases in the transaction have successfully prepared the transaction. Only one gateway instance can participate in an Oracle two-phase commit transaction as the commit point site.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for information about the two-phase commit process.

To enable the COMMIT_CONFIRM transaction mode, perform the following tasks:

1. [Create a Recovery Account and Password](#)
2. [Create the Transaction Log Table](#)

The log table, called HS_TRANSACTION_LOG, is where two-phase commit transactions are recorded.

Create a Recovery Account and Password

For the gateway to recover distributed transactions, a recovery account and password must be set up in the Teradata database. By default, both the user name of the account and the password are RECOVER. The name of the account can be changed with the gateway initialization parameter HS_FDS_RECOVERY_ACCOUNT. The account password can be changed with the gateway initialization parameter HS_FDS_RECOVERY_PWD.

Note: Oracle recommends that you do not use the default value RECOVER for the user name and password. Moreover, storing plain-text as user name and password in the initialization file is not a good security policy. There is now a utility called dg4pwd, that should be used for encryption. Refer to Section 4.2.3, 'Encrypting Initialization parameters' in the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for further details.

1. Set up a user account in the Teradata database. Both the user name and password must be a valid Teradata user name and password.
2. In the initialization parameter file, set the following gateway initialization parameters:
 - HS_FDS_RECOVERY_ACCOUNT to the user name of the Teradata user account you set up for recovery.
 - HS_FDS_RECOVERY_PWD to the password of the Teradata user account you set up for recovery.

See Also: ["Customize the Initialization Parameter File"](#) on page 7-1 for information about editing the initialization parameter file. For information about HS_FDS_RECOVERY_ACCOUNT and HS_FDS_RECOVERY_PWD, see [Appendix C, "Initialization Parameters"](#).

Create the Transaction Log Table

When configuring the gateway for two-phase commit, a table must be created in the Teradata database for logging transactions. The gateway uses the transaction log table to check the status of failed transactions that were started at the Teradata database by the gateway and registered in the table.

Note: Updates to the transaction log table cannot be part of an Oracle distributed transaction.

Note: The information in the transaction log table is required by the recovery process and must not be altered. The table must be used, accessed, or updated only by the gateway.

The table, called HS_TRANSACTION_LOG, consists of two columns, GLOBAL_TRAN_ID, data type CHAR (64) and TRAN_COMMENT, data type CHAR (255).

You can use another name for the log table, other than HS_TRANSACTION_LOG, by specifying the other name using the HS_FDS_TRANSACTION_LOG initialization parameter.

See Also: [Appendix C, "Initialization Parameters"](#) for information about the HS_FDS_TRANSACTION_LOG initialization parameter.

Create the transaction log table in the user account you created in ["Create a Recovery Account and Password"](#) on page 7-8. Because the transaction log table is used to record the status of a gateway transaction, the table must reside at the database where the Teradata update takes place. Also, the transaction log table must be created under the owner of the recovery account.

Note: To utilize the transaction log table, users of the gateway must be granted privileges on the table.

To create a transaction log table use the dg4tera_tx.sql script, located in the directory \$ORACLE_HOME/dg4tera/admin, where \$ORACLE_HOME is the directory under which the gateway is installed.

Encrypt Gateway Initialization Parameter Values

The gateway uses user IDs and passwords to access the information in the remote database. Some user IDs and passwords must be defined in the gateway initialization file to handle functions such as resource recovery. In the current security conscious environment, having plain-text passwords that are accessible in the initialization file is deemed insecure. The dg4pwd encryption utility has been added as part of Heterogeneous Services to help make this more secure. This utility is accessible by this

gateway. The initialization parameters which contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using this utility.

Configure the Gateway to Access Multiple Teradata Databases

The tasks for configuring the gateway to access multiple Teradata databases are similar to the tasks for configuring the gateway for a single database. The configuration example assumes the following:

- The gateway is installed and configured with the default SID of `dg4tera`
- The `ORACLE_HOME` environment variable is set to the directory where the gateway is installed.
- The gateway is configured for one Teradata database named `db1`.
- Two Teradata databases named `db2` and `db3` on a host with IP Address `204.179.79.15` are being added.

Multiple Teradata Databases Example: Configuring the Gateway

Choose One System ID for Each Teradata Database

A separate instance of the gateway is needed for each Teradata database. Each instance needs its own gateway System ID (SID). For this example, the gateway SIDs are chosen for the instances that access the Teradata databases:

- `dg4tera2` for the gateway accessing database `db2`.
- `dg4tera3` for the gateway accessing database `db3`.

Create Two Initialization Parameter Files

Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file:

`$ORACLE_HOME/dg4tera/admin/initdg4tera.ora`, twice, naming one with the gateway SID for `db2` and the other with the gateway SID for `db3`:

```
$ cd $ORACLE_HOME/dg4tera/admin
$ cp initdg4tera.ora initdg4tera2.ora
$ cp initdg4tera.ora initdg4tera3.ora
```

Change the value of the `HS_FDS_CONNECT_INFO` parameter in the new files.

For `initdg4tera2.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:1025/db2
```

For `initdg4tera3.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:1025/db3
```

Note: If you have multiple gateway SIDs for the same Teradata database because you want to use different gateway parameter settings at different times, follow the same procedure. You create several initialization parameter files, each with different SIDs and different parameter settings.

Multiple Teradata Databases Example: Configuring Oracle Net Listener

Add Entries to listener.ora

Add two new entries to the Oracle Net Listener configuration file, `listener.ora`. You must have an entry for each gateway instance, even when multiple gateway instances access the same database.

The following example shows the entry for the original installed gateway first, followed by the new entries.

```
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4tera)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4tera)

    (ENVS=LD_LIBRARY_PATH=oracle_home_directory/lib:teradata_client_library_
directory:/usr/lib)
  )
  (SID_DESC=
    (SID_NAME=dg4tera2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4tera)

    (ENVS=LD_LIBRARY_PATH=oracle_home_directory/lib:teradata_client_library_
directory:/usr/lib)
  )
  (SID_DESC=
    (SID_NAME=dg4tera3)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4tera)

    (ENVS=LD_LIBRARY_PATH=oracle_home_directory/lib:teradata_client_library_
directory:/usr/lib)
  )
)
```

where, `oracle_home_directory` is the directory where the gateway resides.

Note: For HP-UX PA-RISC, the `envs` parameter also needs to be set. Refer to "[Syntax of listener.ora File Entries](#)" on page 7-3 for more information about adding the `envs` parameter.

Multiple Teradata Databases Example: Stopping and Starting the Oracle Net Listener

If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

Multiple Teradata Databases Example: Configuring Oracle Database for Gateway Access

Configuring Oracle Net for Multiple Gateway Instances

Add two connect descriptor entries to the `tnsnames.ora` file. You must have an entry for each gateway instance, even if the gateway instances access the same database.

The following Teradata example shows the entry for the original installed gateway first, followed by the two entries for the new gateway instances:

```
old_db_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4tera) )
  (HS=OK) )
new_db2_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4tera2) )
  (HS=OK) )
new_db3_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4tera3) )
  (HS=OK) )
```

The value for `PORT` is the TCP/IP port number of the Oracle Net Listener that is listening for the gateway. The number can be found in the `listener.ora` file used by the Oracle Net Listener. The value for `HOST` is the name of the machine on which the gateway is running. The name also can be found in the `listener.ora` file used by the Oracle Net Listener.

Multiple Teradata Databases Example: Accessing Teradata Data

Enter the following to create a database link for the `dg4tera2` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK TERA2 CONNECT TO
  2 "user2" IDENTIFIED BY "password2" USING 'new_db2_using';
```

Enter the following to create a database link for the `dg4tera3` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK TERA3 CONNECT TO
  2 "user3" IDENTIFIED BY "password3" USING 'new_db3_using';
```

After the database links are created you can verify the connection to the new Teradata databases, as in the following:

```
SQL> SELECT * FROM ALL_USERS@TERA2;
```

```
SQL> SELECT * FROM ALL_USERS@TERA3;
```

Part V Part Installing

and Configuring Oracle Database Gateway for SQL Server

Part V, "Installing and Configuring Oracle Database Gateway for SQL Server" describes how to install and configure Oracle Database Gateway for SQL Server on UNIX based platforms.

It contains the following chapters:

- Chapter 8, "Installing Oracle Database Gateway for SQL Server"
- Chapter 9, "Configuring Oracle Database Gateway for SQL Server"

Installing Oracle Database Gateway for SQL Server

This chapter provides information about the hardware and software requirements and the installation procedure for Oracle Database Gateway for Microsoft SQL Server.

To install the gateway, follow these steps:

1. Ensure that the system meets all of the hardware and software requirements specified in ["System Requirements for Oracle Database Gateway for SQL Server"](#) on page 8-1.
2. Run the Oracle Universal Installer.

See ["Step Through the Oracle Universal Installer"](#) on page 8-5 for more information about running the Oracle Universal Installer

Oracle Universal Installer is a menu-driven utility that guides you through the installation of the gateway by prompting you with action items. The action items and the sequence in which they appear depend on your platform.

See [Table 8-3](#) for a description of the installation procedure of Oracle Database Gateway for SQL Server.

System Requirements for Oracle Database Gateway for SQL Server

This section provides information about the hardware and software requirements for the gateway. It contains the following sections:

- ["Hardware Requirements"](#) on page 8-1
- ["Software Requirements"](#) on page 8-3

Hardware Requirements

Table 2-1 shows the minimum hardware requirements for Oracle Database Gateway for SQL Server.

Table 8–1 Hardware requirements for Oracle Database Gateway for SQL Server

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for HP-UX Itanium	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Temporary Disk Space	400 MB	400 MB	400 MB	400 MB	400 MB	400 MB
Disk Space	1.5 GB	1.5 GB	1.5 GB	750 MB	750 MB	750 MB
Physical Memory*	512 MB	512 MB	512 MB	512 MB	512 MB	512 MB
Swap Space	1 GB	1 GB	1 GB	1 GB	1 GB	1 GB
Processor	IBM RS/6000 AIX-Based System Processor	HP 9000 Series 700 or 800 processor for hp-ux 11.0	HP Itanium processor for hp-ux 11	Sun Solaris Operating System (SPARC) Processor	x86	x86_64

* The minimum swap space is 1 GB (or twice the size of RAM). On systems with 2 GB or more of RAM, the swap space can be between one and two times the size of RAM. On AIX systems with 1 GB or more of memory, do not increase the swap space more than 2 GB.

Checking the Hardware Requirements

To ensure that the system meets the minimum requirements, follow these steps:

1. To determine the physical RAM size, enter one of the following commands:

Operating System	Command
AIX	<code># /usr/sbin/lsattr -E -l sys0 -a realmem</code>
HP-UX PA-RISC	<code># /usr/sbin/dmesg grep "Physical:"</code>
HP-UX Itanium	<code># /usr/contrib/bin/machinfo grep -i Memory</code>
Solaris (SPARC)	<code># /usr/sbin/prtconf grep "Memory size"</code>
Linux x86	<code># grep MemTotal /proc/meminfo</code>
Linux x86 64 bit	<code># grep MemTotal /proc/meminfo</code>

If the size of the physical RAM installed in the system is less than the required size, you must install more memory before continuing.

2. To determine the size of the configured swap space, enter one of the following commands:

Operating System	Command
AIX	<code># /usr/sbin/lspcs -a</code>
HP-UX PA-RISC	<code># /usr/sbin/swapinfo -a</code>
HP-UX Itanium	<code># /usr/sbin/swapinfo -a</code>
Solaris (SPARC)	<code># /usr/sbin/swap -s</code>
Linux x86	<code># grep SwapTotal /proc/meminfo</code>
Linux x86 64 bit	<code># grep SwapTotal /proc/meminfo</code>

If necessary, see your operating system documentation for information about how to configure additional swap space.

3. To determine the amount of disk space available in the `/tmp` directory enter the following commands:

Operating System	Command
AIX	# <code>df -k /tmp</code>
HP-UX PA-RISC	# <code>df -k /tmp</code>
HP-UX Itanium	# <code>bdf /tmp</code>
Solaris (SPARC)	# <code>df -k /tmp</code>
Linux x86	# <code>df -k /tmp</code>
Linux x86 64 bit	# <code>df -k /tmp</code>

4. To determine the amount of disk space available on the system enter the following commands:

Operating System	Command
AIX	# <code>df -k</code>
HP-UX PA-RISC	# <code>df -k</code>
HP-UX Itanium	# <code>bdf</code>
Solaris (SPARC)	# <code>df -k</code>
Linux x86	# <code>df -k</code>
Linux x86 64 bit	# <code>df -k</code>

Software Requirements

The following section describes the minimum software requirements for Oracle Database Gateway for SQL Server.

Operating System

Table 8–2 shows the minimum operating system version required for Oracle Database Gateway for SQL Server. If your operating system is lower than the minimum requirements, upgrade your operating system to meet the specified levels.

Table 8–2 Operating Systems version for Oracle Database Gateway for SQL Server

Operating System	Version
AIX	AIX 5L version 5.3, Maintenance level 02 or higher
HP-UX PA-RISC	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
HP-UX Itanium	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
Solaris (SPARC)	Solaris 9 Update 6 or higher or Solaris 10
Linux x86 Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 Suse	SUSE Linux Enterprise Server 10.0

Table 8–2 (Cont.) Operating Systems version for Oracle Database Gateway for SQL

Operating System	Version
Linux x86 64 bit Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 64 bit Suse	SUSE Linux Enterprise Server 10.0
Oracle Enterprise Linux x86	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0
Oracle Enterprise Linux x86 64 bit	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0

Checking the Software Requirements

To ensure that the system meets the minimum requirements, follow these steps:

- To determine which version of AIX is installed, enter the following command:

```
# oslevel -r
```
- To determine which version of HP-UX PA-RISC is installed, enter the following command:

```
# uname -a
```
- To determine which version of HP-UX Itanium is installed, enter the following command:

```
# uname -a
```
- To determine which version of Solaris Operating System (SPARC) is installed, enter the following command:

```
# uname -r
```
- To determine which distribution and version of Linux x86 is installed, enter the following command:

```
# cat /etc/issue
```
- To determine which distribution and version of Linux x86 64 bit is installed, enter the following command:

```
# cat /proc/version
```

Certified Configuration

The gateway supports SQL Server. For the latest versions supported refer to the OTN Web site:

<http://www.oracle.com/technology/products/gateways/pdf/certmatrix10g.pdf>

Step Through the Oracle Universal Installer

Table 8–3 describes the installation procedure for Oracle Database Gateway for SQL Server

Table 8–3 The Oracle Universal Installer: Steps for Installing the Gateway

Screen	Response
Oracle Universal Installer: Welcome	Click Next .
Oracle Universal Installer: Specify Home Details	<p>Specify a name for the installation in the Name field. You can also choose not to edit the default setting of the Name field of the Specify Home Details screen.</p> <p>The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to <i>ORACLE_HOME</i>. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation media, the Oracle Universal Installer displays the Available Products screen.</p>
Oracle Universal Installer: Available Product Components	<p>a. Select Oracle Database Gateway for SQL Server 11.1.0.6.0.</p> <p>b. Click Next.</p>
Oracle Database Gateway for SQL Server	<p>SQL Server Database Server Host Name - Specify the host name of the machine hosting the MS SQL Server database.</p> <p>SQL Server Database Server Port number - Specify the port number of the SQL Server database server</p> <p>SQL Server Database Name - Specify the SQL Server database name</p> <p>Click Next to continue.</p>
Oracle Universal Installer: Summary	The Installation Summary screen enables you to review a tree list of options and components for this installation. Click Install to start installation.
Oracle Net Configuration Assistant: Welcome	Click Cancel .
Oracle Net Configuration Assistant:	Click Yes .
Oracle Universal Installer: Configuration Tools	Click Exit .
Exit	The final screen of the Oracle Universal Installer is the End of Installation screen. Click Exit to exit the installer.

Configuring Oracle Database Gateway for SQL Server

After installing the gateway, perform the following tasks to configure Oracle Database Gateway for SQL Server:

1. [Configure the Gateway Initialization Parameter File](#)
2. [Configure Oracle Net for the Gateway](#)
3. [Configure the Oracle Database for Gateway Access](#)
4. [Create Database Links](#)
5. [Configure Two-Phase Commit](#)
6. [Create SQL Server Views for Data Dictionary Support](#)
7. [Encrypt Gateway Initialization Parameter Values](#)
8. [Configure the Gateway to Access Multiple SQL Server Databases](#)

Configure the Gateway Initialization Parameter File

Perform the following tasks to configure the gateway initialization parameter file.

1. [Choose a System Identifier for the Gateway](#)
2. [Customize the Initialization Parameter File](#)

Choose a System Identifier for the Gateway

The gateway system identifier (SID) is an alphanumeric character string that identifies a gateway instance. You need one gateway instance, and therefore one gateway SID, for each SQL Server database you are accessing. The SID is used as part of the file name for the initialization parameter file. The default SID is `dg4msql`.

You can define a gateway SID, but using the default of `dg4msql` is easier because you do not need to change the initialization parameter file name. However, if you want to access two SQL Server databases, you need two gateway SIDs, one for each instance of the gateway. If you have only one SQL Server database and want to access it sometimes with one set of gateway parameter settings, and other times with different gateway parameter settings, then you will need multiple gateway SIDs for the single SQL Server database.

Customize the Initialization Parameter File

The initialization parameter file must be available when the gateway is started. During installation, the following default initialization parameter file is created:

```
$ORACLE_HOME/dg4msql/admin/initdg4msql.ora
```

Where `$ORACLE_HOME` is the directory under which the gateway is installed.

This initialization file is for the default gateway SID. If you are not using `dg4msql` as the gateway SID, you must rename the initialization parameter file using the SID you chose in the preceding step "[Choose a System Identifier for the Gateway](#)" on page 9-1. This default initialization parameter file is sufficient for starting the gateway, verifying a successful installation, and running the demonstration scripts.

A number of initialization parameters can be used to modify the gateway behavior. Refer to [Appendix C, "Initialization Parameters"](#) for the complete list of initialization parameters that can be set. Changes made to the initialization parameters only take effect in the next gateway session. The most important parameter is the `HS_FDS_CONNECT_INFO` which describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. The syntax for `HS_FDS_CONNECT_INFO` is as follows:

```
HS_FDS_CONNECT_INFO=host_name[:port_number][/[instance_name]][/database_name]
```

Where:

Variable	Description
<i>host_name</i>	is the host name or IP address of the machine hosting the SQL Server database.
<i>port_number</i>	is the port number of the SQL Server database.
<i>instance_name</i>	is the instance of SQL Server running on the machine.
<i>database_name</i>	is the SQL Server Database database name.

Either of the variables *port_number* or *instance_name* can be used, but not both together. Optionally, they both can be omitted. The variable *database_name* is always optional. The slash (/) is required when a particular value is omitted. For example, all of the following entries are valid:

```
HS_FDS_CONNECT_INFO=host_name/instance_name/database_name
HS_FDS_CONNECT_INFO=host_name//database_name
HS_FDS_CONNECT_INFO=host_name:port_name//database_name
HS_FDS_CONNECT_INFO=host_name/instance_name
HS_FDS_CONNECT_INFO=host_name
```

See Also: [Appendix C, "Initialization Parameters"](#) and the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about customizing the initialization parameter file.

Configure Oracle Net for the Gateway

The gateway requires Oracle Net to communicate with the Oracle database. After configuring the gateway, perform the following tasks to configure Oracle Net to work with the gateway:

1. [Configure Oracle Net Listener for the Gateway](#)

2. Stop and Start the Oracle Net Listener for the Gateway

Configure Oracle Net Listener for the Gateway

The Oracle Net Listener listens for incoming requests from the Oracle database. For the Oracle Net Listener to listen for the gateway, information about the gateway must be added to the Oracle Net Listener configuration file, `listener.ora`. This file by default is located in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory under which the gateway is installed.

The following entries must be added to the `listener.ora` file:

- A list of Oracle Net addresses on which the Oracle Net Listener listens
- The executable name of the gateway that the Oracle Net Listener starts in response to incoming connection requests

A sample of the `listener.ora` entry (`listener.ora.sample`) is available in the `$ORACLE_HOME/dg4msql/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Syntax of listener.ora File Entries

The Oracle database communicates with the gateway using Oracle Net and any supported protocol adapters. The following is the syntax of the address on which the Oracle Net Listener listens using the TCP/IP protocol adapter:

```
LISTENER=
  (ADDRESS=
    (PROTOCOL=TCP)
    (HOST=host_name)
    (PORT=port_number) )
```

Where:

Variable	Description
<i>host_name</i>	is the name of the machine on which the gateway is installed.
<i>port_number</i>	specifies the port number used by the Oracle Net Listener. If you have other listeners running on the same machine, then the value of <i>port_number</i> must be different from the other listeners' port numbers.

To direct the Oracle Net Listener to start the gateway in response to incoming connection requests, add an entry to the `listener.ora` file. The syntax for HP-UX PA-RISC slightly different than the other platforms.

Note: You must use the same SID value in the `listener.ora` file and the `tnsnames.ora` file which will be configured in the next step.

For AIX, Solaris SPARC, and Linux:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4msql)
```

```
)
)
```

For HP-UX Itanium:

```
SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(ENVS=LD_LIBRARY_PATH=oracle_home_directory/dg4msql/driver/lib)
(PROGRAM=dg4msql)
)
)
```

For HP-UX PA-RISC:

```
SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(PROGRAM=dg4msql)
(ENVS=SHLIB_PATH=$ORACLE_HOME/lib32)
)
)
```

Where:

Variable	Description
<i>gateway_sid</i>	specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <i>tnsnames.ora</i> file.
<i>oracle_home_directory</i>	specifies the Oracle home directory where the gateway resides.
<i>dg4msql</i>	specifies the executable name of the Oracle Database Gateway for SQL Server.

If you already have an existing Oracle Net Listener, then add the following syntax to *SID_LIST* in the existing *listener.ora* file:

For AIX, Solaris SPARC, and Linux:

```
SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=.
.
)
(SID_DESC=.
.
)
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(PROGRAM=dg4msql)
)
)
```

For HP-UX Itanium:


```

SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=.
    .
  )
  (SID_DESC=.
    .
  )
  (SID_DESC=
    (SID_NAME=gateway_sid)
    (ORACLE_HOME=oracle_home_directory)
    (ENVS=LD_LIBRARY_PATH=oracle_home_directory/dg4msql/driver/lib)
    (PROGRAM=dg4msql)
  )
)

```

For HP-UX PA-RISC:

```

SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=.
    .
  )
  (SID_DESC=.
    .
  )
  (SID_DESC=
    (SID_NAME=gateway_sid)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4msql)
    (ENVS=SHLIB_PATH=$ORACLE_HOME/lib32)
  )
)

```

See Also: *Oracle Net Administrator's Guide* for information about changing the `listener.ora` file.

Stop and Start the Oracle Net Listener for the Gateway

You must stop and restart the Oracle Net Listener to initiate the new settings, as follows:

1. Set the `PATH` environment variable to `$ORACLE_HOME/bin` where `$ORACLE_HOME` is the directory in which the gateway is installed.

For example on the Linux platform, if you have the Bourne or Korn Shell, enter the following:

```

$ PATH=$ORACLE_HOME/bin:$PATH;export PATH
$ LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH; export LD_LIBRARY_PATH

```

If you have the C Shell, enter the following:

```

$ setenv PATH $ORACLE_HOME/bin:$PATH
$ setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH

```

[Table 9–1](#) specifies which parameter value to use for the different platforms:

Table 9–1 Parameter Values for UNIX Based Platforms

Platform	Parameter Value
Solaris (SPARC) 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>

Table 9–1 (Cont.) Parameter Values for UNIX Based Platforms

Platform	Parameter Value
HP-UX PA-RISC	SHLIB_PATH=\$ORACLE_HOME/lib
HP-UX Itanium	LD_LIBRARY_PATH=\$ORACLE_HOME/lib
Linux x86, and Linux x86 64 bit	LD_LIBRARY_PATH=\$ORACLE_HOME/lib
AIX	LIBPATH=\$ORACLE_HOME/lib

- If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

- Check the status of the listener with the new settings, as follows:

```
$ lsnrctl status
```

The following is a partial output from a `lsnrctl` status check:

```
.
.
.
Services Summary...
Service "dg4msql" has 1 instance(s).
  Instance "dg4msql", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

In this example, the service name is `dg4msql` which is the default SID value assigned during installation.

Configure the Oracle Database for Gateway Access

Before you use the gateway to access SQL Server data you must configure the Oracle database to enable communication with the gateway over Oracle Net.

To configure the Oracle database you must add connect descriptors to the `tnsnames.ora` file. By default, this file is in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory in which the Oracle database is installed. You cannot use the Oracle Net Assistant or the Oracle Net Easy Config tools to configure the `tnsnames.ora` file. You must edit the file manually.

A sample of the `tnsnames.ora` entry (`tnsnames.ora.sample`) is available in the `$ORACLE_HOME/dg4msql/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Configuring `tnsnames.ora`

Edit the `tnsnames.ora` file to add a connect descriptor for the gateway. The following is a syntax of the Oracle Net entry using the TCP/IP protocol:

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name)
      (PORT=port_number)
    )
```

```
(CONNECT_DATA=
  (SID=gateway_sid)
(HS=OK))
```

Where:

Variable	Description
<i>connect_descriptor</i>	is the description of the object to connect to as specified when creating the database link, such as dg4msql. Check the sqlnet.ora file for the following parameter setting: <ul style="list-style-type: none"> names.directory_path = (TNSNAMES) Note: The sqlnet.ora file is typically stored in \$ORACLE_HOME/network/admin.
TCP	is the TCP protocol used for TCP/IP connections.
<i>host_name</i>	specifies the machine where the gateway is running.
<i>port_number</i>	matches the port number used by the Oracle Net Listener that is listening for the gateway. The Oracle Net Listener's port number can be found in the listener.ora file used by the Oracle Net Listener. See " Syntax of listener.ora File Entries " on page 9-3.
<i>gateway_sid</i>	specifies the SID of the gateway and matches the SID specified in the listener.ora file of the Oracle Net Listener that is listening for the gateway. See " Configure Oracle Net Listener for the Gateway " on page 9-3 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

See Also: *Oracle Database Administrator's Guide* for information about editing the tnsnames.ora file.

Create Database Links

Any Oracle client connected to the Oracle database can access SQL Server data through the gateway. The Oracle client and the Oracle database can reside on different machines. The gateway accepts connections only from the Oracle database.

A connection to the gateway is established through a database link when it is first used in an Oracle session. In this context, a connection refers to the connection between the Oracle database and the gateway. The connection remains established until the Oracle session ends. Another session or user can access the same database link and get a distinct connection to the gateway and SQL Server database.

Database links are active for the duration of a gateway session. If you want to close a database link during a session, you can do so with the ALTER SESSION statement.

To access the SQL Server, you must create a database link. A public database link is the most common of database links.

```
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO
2 "user" IDENTIFIED BY "password" USING 'tns_name_entry';
```

Where:

Variable	Description
<i>dblink</i>	is the complete database link name.

Variable	Description
<code>tns_name_entry</code>	specifies the Oracle Net connect descriptor specified in the <code>tnsnames.ora</code> file that identifies the gateway

After the database link is created you can verify the connection to the SQL Server database, as follows:

```
SQL> SELECT * FROM DUAL@dblink;
```

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using database links.

Configure Two-Phase Commit

The gateway supports the following transaction capabilities:

- COMMIT_CONFIRM
- READ_ONLY
- SINGLE_SITE

The transaction model is set using the `HS_TRANSACTION_MODEL` initialization parameter. By default, the gateway runs in `COMMIT_CONFIRM` transaction mode. When the SQL Server database is updated by a transaction, the gateway becomes the commit point site. The Oracle database commits the unit of work in the SQL Server database after verifying that all Oracle databases in the transaction have successfully prepared the transaction. Only one gateway instance can participate in an Oracle two-phase commit transaction as the commit point site.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for information about the two-phase commit process.

To enable the `COMMIT_CONFIRM` transaction mode, perform the following tasks:

1. [Create a Recovery Account and Password](#)
2. [Create the Transaction Log Table](#)

The log table, called `HS_TRANSACTION_LOG`, is where two-phase commit transactions are recorded.

Create a Recovery Account and Password

For the gateway to recover distributed transactions, a recovery account and password must be set up in the SQL Server database. By default, both the user name of the account and the password are `RECOVER`. The name of the account can be changed with the gateway initialization parameter `HS_FDS_RECOVERY_ACCOUNT`. The account password can be changed with the gateway initialization parameter `HS_FDS_RECOVERY_PWD`.

Note: Oracle recommends that you do not use the default value RECOVER for the user name and password. Moreover, storing plain-text as user name and password in the initialization file is not a good security policy. There is now a utility called `dg4pwd`, that should be used for encryption. Refer to Section 4.2.3, 'Encrypting Initialization parameters' in the *Oracle Database Heterogeneous Connectivity Administrator's Guide* for further details.

1. Set up a user account in the SQL Server database. Both the user name and password must be a valid SQL Server user name and password.
2. In the initialization parameter file, set the following gateway initialization parameters:
 - HS_FDS_RECOVERY_ACCOUNT to the user name of the SQL Server user account you set up for recovery.
 - HS_FDS_RECOVERY_PWD to the password of the SQL Server user account you set up for recovery.

See Also: ["Customize the Initialization Parameter File"](#) on page 9-2 for information about editing the initialization parameter file. For information about HS_FDS_RECOVERY_ACCOUNT and HS_FDS_RECOVERY_PWD, see [Appendix C, "Initialization Parameters"](#).

Create the Transaction Log Table

When configuring the gateway for two-phase commit, a table must be created in the SQL Server database for logging transactions. The gateway uses the transaction log table to check the status of failed transactions that were started at the SQL Server database by the gateway and registered in the table.

Note: Updates to the transaction log table cannot be part of an Oracle distributed transaction.

Note: The information in the transaction log table is required by the recovery process and must not be altered. The table must be used, accessed, or updated only by the gateway.

The table, called HS_TRANSACTION_LOG, consists of two columns, GLOBAL_TRAN_ID, data type CHAR(64) NOT NULL and TRAN_COMMENT, data type CHAR(255).

You can use another name for the log table, other than HS_TRANSACTION_LOG, by specifying the other name using the HS_FDS_TRANSACTION_LOG initialization parameter.

See Also: [Appendix C, "Initialization Parameters"](#) for information about the HS_FDS_TRANSACTION_LOG initialization parameter.

Create the transaction log table in the user account you created in ["Create a Recovery Account and Password"](#) on page 9-8. Because the transaction log table is used to record the status of a gateway transaction, the table must reside at the database where the

SQL Server update takes place. Also, the transaction log table must be created under the owner of the recovery account.

Note: To utilize the transaction log table, users of the gateway must be granted privileges on the table.

To create a transaction log table use the `dg4msql_tx.sql` script, located in the directory `$ORACLE_HOME/dg4msql/admin` where `$ORACLE_HOME` is the directory under which the gateway is installed. Use `isql` to execute the script, as follows:

```
$ isql -Urecovery_account -Precovery_account_password [-Sserver] -idg4msql_tx.sql
```

Create SQL Server Views for Data Dictionary Support

To enable Oracle data dictionary translation support use the `dg4msql_cvw.sql` script, located in the directory `$ORACLE_HOME/dg4msql/admin` where `$ORACLE_HOME` is the directory under which the gateway is installed. You must run this script on each SQL Server database that you want to access through the gateway. Use `isql` to execute the script, as follows:

```
$ isql -Usa_user -Psa_pwd [-Sserver] [-ddatabase] -e -i dg4msql_cvw.sql
```

where `sa_user` and `sa_pwd` are the SQL Server system administrator user ID and password respectively.

Encrypt Gateway Initialization Parameter Values

The gateway uses user IDs and passwords to access the information in the remote database. Some user IDs and passwords must be defined in the gateway initialization file to handle functions such as resource recovery. In the current security conscious environment, having plain-text passwords that are accessible in the initialization file is deemed insecure. The `dg4pwd` encryption utility has been added as part of Heterogeneous Services to help make this more secure. This utility is accessible by this gateway. The initialization parameters which contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using this utility.

Configure the Gateway to Access Multiple SQL Server Databases

The tasks for configuring the gateway to access multiple SQL Server databases are similar to the tasks for configuring the gateway for a single database. The configuration example assumes the following:

- The gateway is installed and configured with the default SID of `dg4msql`
- The `ORACLE_HOME` environment variable is set to the directory where the gateway is installed
- The gateway is configured for one SQL Server database named `db1`
- Two SQL Server databases named `db2` and `db3` on a host with IP Address 204.179.79.15 are being added

Multiple SQL Server Databases Example: Configuring the Gateway

Choose One System ID for Each SQL Server Database

A separate instance of the gateway is needed for each SQL Server database. Each instance needs its own gateway System ID (SID). For this example, the gateway SIDs are chosen for the instances that access the SQL Server databases:

- dg4msql2 for the gateway accessing database db2
- dg4msql3 for the gateway accessing database db3

Create Two Initialization Parameter Files

Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file, `$ORACLE_HOME/dg4msql/admin/initdg4msql.ora`, twice, naming one with the gateway SID for db2 and the other with the gateway SID for db3:

```
$ cd $ORACLE_HOME/dg4msql/admin
$ cp initdg4msql.ora initdg4msql2.ora
$ cp initdg4msql.ora initdg4msql3.ora
```

Change the value of the `HS_FDS_CONNECT_INFO` parameter in the new files.

For `initdg4msql2.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:1433//db2
```

For `initdg4msql3.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=204.179.79.15:1433//db3
```

Note: If you have multiple gateway SIDs for the same SQL Server database because you want to use different gateway parameter settings at different times, follow the same procedure. You create several initialization parameter files, each with different SIDs and different parameter settings.

Multiple SQL Server Databases Example: Configuring Oracle Net Listener

Add Entries to listener.ora

Add two new entries to the Oracle Net Listener configuration file, `listener.ora`. You must have an entry for each gateway instance, even when multiple gateway instances access the same database.

The following example shows the entry for the original installed gateway first, followed by the new entries:

```
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4msql)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4msql)
  )
  (SID_DESC=
    (SID_NAME=dg4msql2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4msql)
```

```
)
(SID_DESC=
  (SID_NAME=dg4msql3)
  (ORACLE_HOME=oracle_home_directory)
  (PROGRAM=dg4msql)
)
)
```

where, *oracle_home_directory* is the directory where the gateway resides.

Note: For HP-UX PA-RISC, the *envs* parameter also needs to be set. Refer to "[Syntax of listener.ora File Entries](#)" on page 9-3 for more information about adding the *envs* parameter.

Multiple SQL Server Databases Example: Stopping and Starting the Oracle Net Listener

If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

Multiple SQL Server Databases Example: Configuring Oracle Database for Gateway Access

Configuring Oracle Net for Multiple Gateway Instances

Add two connect descriptor entries to the `tnsnames.ora` file. You must have an entry for each gateway instance, even if the gateway instances access the same database.

The following SQL Server example shows the entry for the original installed gateway first, followed by the two entries for the new gateway instances:

```
old_db_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4msql) )
  (HS=OK) )
new_db2_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4msql2) )
  (HS=OK) )
new_db3_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4msql3) )
```


(HS=OK)

The value for `PORT` is the TCP/IP port number of the Oracle Net Listener that is listening for the gateway. The number can be found in the `listener.ora` file used by the Oracle Net Listener. The value for `HOST` is the name of the machine on which the gateway is running. The name also can be found in the `listener.ora` file used by the Oracle Net Listener.

Multiple SQL Server Databases Example: Accessing SQL Server Data

Enter the following to create a database link for the `dg4msql2` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK MSQ2 CONNECT TO
  2 "user2" IDENTIFIED BY "password2" USING 'new_db2_using';
```

Enter the following to create a database link for the `dg4msql3` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK MSQ3 CONNECT TO
  2 "user3" IDENTIFIED BY "password3" USING 'new_db3_using';
```

After the database links are created you can verify the connection to the new SQL Server databases, as in the following:

```
SQL> SELECT * FROM ALL_USERS@MSQ2;
```

```
SQL> SELECT * FROM ALL_USERS@MSQ3;
```


Part VI

Installing and Configuring Oracle Database Gateway for ODBC

Part VI, "Installing and Configuring Oracle Database Gateway for ODBC" describes how to install and configure Oracle Database Gateway for ODBC on UNIX based platforms.

It contains the following chapters:

- [Chapter 10, "Installing Oracle Database Gateway for ODBC"](#)
- [Chapter 11, "Configuring Oracle Database Gateway for ODBC"](#)

Installing Oracle Database Gateway for ODBC

This chapter provides information about the hardware and software requirements and the installation procedure for Oracle Database Gateway for ODBC.

To install the gateway, follow these steps:

1. Ensure that the system meets all of the hardware and software requirements specified in ["System Requirements for Oracle Database Gateway for ODBC"](#) on page 10-1

2. Run the Oracle Universal Installer.

See ["Step Through the Oracle Universal Installer"](#) on page 10-5 for more information about running the Oracle Universal Installer

Oracle Universal Installer is a menu-driven utility that guides you through the installation of the gateway by prompting you with action items. The action items and the sequence in which they appear depend on your platform.

See [Table 10-3](#) for a description of the installation procedure of Oracle Database Gateway for ODBC

System Requirements for Oracle Database Gateway for ODBC

This section provides information about the hardware and software requirements for the gateway. It contains the following sections:

- ["Hardware Requirements"](#) on page 10-1
- ["Software Requirements"](#) on page 10-3

Hardware Requirements

[Table 10-1](#) shows the minimum hardware requirements for Oracle Database Gateway for ODBC.

Table 10–1 Hardware requirements for Oracle Database Gateway for ODBC

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for HP-UX Itanium	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit**
Temporary Disk Space	400 MB	400 MB	400 MB	400 MB	400 MB	400 MB
Disk Space	1.5 GB	1.5 GB	1.5 GB	750 MB	750 MB	750 MB
Physical Memory*	512 MB	512 MB	512 MB	512 MB	512 MB	512 MB
Swap Space	1 GB	1 GB	1 GB	1 GB	1 GB	1 GB
Processor	IBM RS/6000 AIX-Based System Processor	HP 9000 Series 700 or 800 processor for hp-ux 11.0	HP Itanium processor for hp-ux 11	Sun Solaris Operating System (SPARC) Processor	x86	x86_64

* The minimum swap space is 1 GB (or twice the size of RAM). On systems with 2 GB or more of RAM, the swap space can be between one and two times the size of RAM. On AIX systems with 1 GB or more of memory, do not increase the swap space more than 2 GB.

**Database Gateway for ODBC on Linux x86-64 is now a 64-bit application that requires the use of a 64-bit third party ODBC Driver.

Checking the Hardware Requirements

To ensure that the system meets the minimum requirements, follow these steps:

1. To determine the physical RAM size, enter one of the following commands:

Operating System	Command
AIX	<code># /usr/sbin/lsattr -E -l sys0 -a realmem</code>
HP-UX PA-RISC	<code># /usr/sbin/dmesg grep "Physical:"</code>
HP-UX Itanium	<code># /usr/contrib/bin/machinfo grep -i Memory</code>
Solaris (SPARC)	<code># /usr/sbin/prtconf grep "Memory size"</code>
Linux x86	<code># grep MemTotal /proc/meminfo</code>
Linux x86 64 bit	<code># grep MemTotal /proc/meminfo</code>

If the size of the physical RAM installed in the system is less than the required size, you must install more memory before continuing.

2. To determine the size of the configured swap space, enter one of the following commands:

Operating System	Command
AIX	<code># /usr/sbin/lsps -a</code>
HP-UX PA-RISC	<code># /usr/sbin/swapinfo -a</code>
HP-UX Itanium	<code># /usr/sbin/swapinfo -a</code>
Solaris (SPARC)	<code># /usr/sbin/swap -s</code>
Linux x86	<code># grep SwapTotal /proc/meminfo</code>

Operating System	Command
Linux x86 64 bit	# grep SwapTotal /proc/meminfo

If necessary, see your operating system documentation for information about how to configure additional swap space.

- To determine the amount of disk space available in the /tmp directory enter the following commands:

Operating System	Command
AIX	# df -k /tmp
HP-UX PA-RISC	# df -k /tmp
HP-UX Itanium	# bdf /tmp
Solaris (SPARC)	# df -k /tmp
Linux x86	# df -k /tmp
Linux x86 64 bit	# df -k /tmp

- To determine the amount of disk space available on the system enter the following commands:

Operating System	Command
AIX	# df -k
HP-UX PA-RISC	# df -k
HP-UX Itanium	# bdf
Solaris (SPARC)	# df -k
Linux x86	# df -k
Linux x86 64 bit	# df -k

Software Requirements

The following section describes the minimum software requirements for Oracle Database Gateway for ODBC.

Operating System

Table 10-3 shows the minimum operating system version required for Oracle Database Gateway for ODBC. If your operating system is lower than the minimum requirements, upgrade your operating system to meet the specified levels.

Table 10-2 Operating Systems version for Oracle Database Gateway for ODBC

Operating System	Version
AIX	AIX 5L version 5.3, Maintenance level 02 or higher
HP-UX PA-RISC	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
HP-UX Itanium	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
Solaris (SPARC)	Solaris 9 Update 6 or higher or Solaris 10

Table 10–2 (Cont.) Operating Systems version for Oracle Database Gateway for ODBC

Operating System	Version
Linux x86 Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 Suse	SUSE Linux Enterprise Server 10.0
Linux x86 64 bit Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 64 bit Suse	SUSE Linux Enterprise Server 10.0
Oracle Enterprise Linux x86	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0
Oracle Enterprise Linux x86 64 bit	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0

Checking the Software Requirements

To ensure that the system meets the minimum requirements, follow these steps:

- To determine which version of AIX is installed, enter the following command:

```
# oslevel -r
```
- To determine which version of HP-UX PA-RISC is installed, enter the following command:

```
# uname -a
```
- To determine which version of HP-UX Itanium is installed, enter the following command:

```
# uname -a
```
- To determine which version of Solaris Operating System (SRPARC) is installed, enter the following command:

```
# uname -r
```
- To determine which distribution and version of Linux x86 is installed, enter the following command:

```
# cat /etc/issue
```
- To determine which distribution and version of Linux x86 64 bit is installed, enter the following command:

```
# cat /proc/version
```

Certified Configuration

For the latest certified configuration refer to the OTN Web site:

<http://www.oracle.com/technology/products/gateways/pdf/certmatrix10g.pdf>

Step Through the Oracle Universal Installer

Table 10–3 describes the installation procedure for Oracle Database Gateway for ODBC.

Table 10–3 The Oracle Universal Installer: Steps for Installing Oracle Database Gateway for ODBC

Screen	Response
Oracle Universal Installer: Welcome	Click Next .
Oracle Universal Installer: File Locations	<p>The Source section of the screen is where you specify the source location that the Oracle Universal Installer must use to install the Oracle Database Gateway for ODBC. You need not edit the file specification in the Path field. The default setting for this field points to the installer file on your gateway installation media.</p> <p>The Path field in the Destination section of the File Locations screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to <code>ORACLE_HOME</code>. After you set the fields in the File Locations screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen.</p>
Oracle Universal Installer: Available Product Components	<p>a. Select Oracle Database Gateway for ODBC 11.1.0.6.0.</p> <p>b. Click Next.</p>
Oracle Universal Installer: Summary	The Installation Summary screen enables you to review a tree list of options and components for this installation. Click Install to start installation.
Oracle Net Configuration Assistant: Welcome	Click Cancel
Oracle Net Configuration Assistant:	Click Yes
Oracle Universal Installer: Configuration Tools	Click Exit
Exit	The final screen of the Oracle Universal Installer is the End of Installation screen. Click Exit to exit the installer.

Configuring Oracle Database Gateway for ODBC

After installing the gateway and the ODBC driver for the non-Oracle system, perform the following tasks to configure Oracle Database Gateway for ODBC:

1. [Configure the Gateway Initialization Parameter File](#)
2. [Configure Oracle Net for the Gateway](#)
3. [Configure the Oracle Database for Gateway Access](#)
4. [Create Database Links](#)
5. [Encrypt Gateway Initialization Parameter Values](#)
6. [Configure the Gateway to Access Multiple ODBC Data Sources](#)

Configure the Gateway Initialization Parameter File

Perform the following tasks to configure the gateway initialization file:

1. [Create the Initialization Parameter File](#)
2. [Set the Initialization Parameter Values](#)

Create the Initialization Parameter File

You must create an initialization file for your Oracle Database Gateway for ODBC. Oracle supplies a sample initialization file, `initdg4odbc.ora`. The sample file is stored in the `$ORACLE_HOME/hs/admin` directory.

To create an initialization file for the ODBC gateway, copy the sample initialization file and rename it to `initsid.ora`, where *sid* is the system identifier (SID) you want to use for the instance of the non-Oracle system to which the gateway connects.

The gateway system identifier (SID) is an alphanumeric character string that identifies a gateway instance. You need one gateway instance, and therefore one gateway SID, for each ODBC source you are accessing.

If you want to access two ODBC sources, you need two gateway SIDs, one for each instance of the gateway. If you have only one ODBC source but want to access it sometimes with one set of gateway parameter settings, and other times with different gateway parameter settings, then you will need multiple gateway SIDs for the single ODBC source. The SID is used as part of the file name for the initialization parameter file.

Set the Initialization Parameter Values

After the initialization file has been created, you must set the initialization parameter values. A number of initialization parameters can be used to modify the gateway behavior. You must set the `HS_FDS_CONNECT_INFO` and the `HS_FDS_SHAREABLE_NAME` initialization parameters. Other initialization parameters have defaults or are optional. You can use the default values and omit the optional parameters, or you can specify the parameters with values tailored for your installation. Refer to [Appendix C, "Initialization Parameters"](#) for the complete list of initialization parameters that can be set. Changes made to the initialization parameters only take effect in the next gateway session.

The `HS_FDS_CONNECT_INFO` initialization parameter specifies the information required for connecting to the non-Oracle system. Set the `HS_FDS_CONNECT_INFO` as follows:

```
HS_FDS_CONNECT_INFO=dsn_value
```

where *dsn_value* is the data source name configured in the `odbc.ini` file

The `HS_FDS_SHAREABLE_NAME` initialization parameter specifies the full path of the ODBC driver manager. Set the `HS_FDS_SHAREABLE_NAME` as follows:

```
HS_FDS_SHAREABLE_NAME=full_path_of_odbc_driver
```

where *full_path_of_odbc_driver* is the full path to the ODBC driver manager

Note: Before deciding whether to accept the default values or to change them, see [Appendix C, "Initialization Parameters"](#) for detailed information about all the initialization parameters.

Example: Setting Initialization Parameter Values

The following is an example of an `odbc.ini` file that uses DataDirect Technologies SQLServer ODBC driver. The ODBC driver is installed in `$ODBCHOME`, which is the `/opt/odbc520` directory.

```
[ODBC Data Sources]
SQLServerWP=DataDirect 5.20 SQL Server Wire Protocol

[SQLServerWP]
Driver=/opt/odbc520/lib/ivmsss18.so
Description=DataDirect 5.20 SQL Server Wire Protocol
Database=oratst
LogonID=TKHUSER
Password=TKHUSER
Address=sqlserver-pc,1433
QuotedId=Yes
AnsiNPW=No

[ODBC]
Trace=0
TraceFile=/opt/odbc520/odbctrace.out
TraceDll=/opt/odbc520/lib/odbctrac.so
InstallDir=/opt/odb520
ConversionTableLocation=/opt/odbc520/tables
UseCursorLib=0
```

To configure the Gateway for ODBC to use this driver, the following lines are required in `initSID.ora`:

```
HS_FDS_CONNECT_INFO=SQLServerWP
HS_FDS_SHAREABLE_NAME=/opt/odbc520/lib/libodbc.so
set ODBCINI=/opt/odbc/odbc.ini
```

If the ODBC driver you are using requires you to set some environment variables then you can either set them in the initialization file or in the environment.

The `HS_FDS_CONNECT_INFO` initialization parameter value must match the ODBC data source name in the `odbc.ini` file.

Note: If the ODBC driver supports Quoted Identifiers or Delimited Identifiers it should be turned on.

Configure Oracle Net for the Gateway

The gateway requires Oracle Net to communicate with the Oracle database. After configuring the gateway, perform the following tasks to configure Oracle Net to work with the gateway:

1. [Configure Oracle Net Listener for the Gateway](#)
2. [Stop and Start the Oracle Net Listener for the Gateway](#)

Configure Oracle Net Listener for the Gateway

The Oracle Net Listener listens for incoming requests from the Oracle database. For the Oracle Net Listener to listen for the gateway, information about the gateway must be added to the Oracle Net Listener configuration file, `listener.ora`. This file by default is located in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory under which the gateway is installed.

The following entries must be added to the `listener.ora` file:

- A list of Oracle Net addresses on which the Oracle Net Listener listens
- The executable name of the gateway that the Oracle Net Listener starts in response to incoming connection requests

A sample of the `listener.ora` entry (`listener.ora.sample`) is available in the `$ORACLE_HOME/dg4odbc/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Syntax of listener.ora File Entries

The Oracle database communicates with the gateway using Oracle Net and any supported protocol adapters. The following is the syntax of the address on which the Oracle Net Listener listens using the TCP/IP protocol adapter:

```
LISTENER=
  (ADDRESS=
    (PROTOCOL=TCP)
    (HOST=host_name)
    (PORT=port_number))
```

Where:

Variable	Description
<code>host_name</code>	is the name of the machine on which the gateway is installed.

Variable	Description
<i>port_number</i>	specifies the port number used by the Oracle Net Listener. If you have other listeners running on the same machine, then the value of <i>port_number</i> must be different from the other listeners' port numbers.

To direct the Oracle Net Listener to start the gateway in response to incoming connection requests, add an entry to the `listener.ora` file. The syntax for HP-UX PA-RISC slightly different than the other platforms.

Note: You must use the same SID value in the `tnsnames.ora` file and the `listener.ora` file.

For Linux:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4odbc)
      (ENVS=LD_LIBRARY_PATH=odbc_library_dir:$ORACLE_HOME/lib)
    )
  )
```

For HP-UX PA-RISC:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4odbc)
      (ENVS=SHLIB_PATH=odbc_library_dir:$ORACLE_HOME/lib32)
    )
  )
```

Where:

Variable	Description
<i>gateway_sid</i>	specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <code>tnsnames.ora</code> file.
<i>oracle_home_directory</i>	specifies the Oracle home directory where the gateway resides.
<i>odbc_library_dir</i>	specifies the ODBC driver library path
<i>dg4odbc</i>	specifies the executable name of the Oracle Database Gateway for ODBC.

If you already have an existing Oracle Net Listener, then add the following syntax to `SID_LIST` in the existing `listener.ora` file:

For Linux:

```
SID_LIST_LISTENER=
  (SID_LIST=
```

```

(SID_DESC=.
.
)
(SID_DESC=.
.
)
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(PROGRAM=dg4odbc)
(ENVS=LD_LIBRARY_PATH=odbc_library_dir:$ORACLE_HOME/lib)
)
)

```

For HP-UX PA-RISC:

```

SID_LIST_LISTENER=
(SID_LIST=
(SID_DESC=.
.
)
(SID_DESC=.
.
)
(SID_DESC=
(SID_NAME=gateway_sid)
(ORACLE_HOME=oracle_home_directory)
(PROGRAM=dg4odbc)
(ENVS=SHLIB_PATH=odbc_library_dir:$ORACLE_HOME/lib32)
)
)

```

See Also: *Oracle Net Administrator's Guide* for information about changing the `listener.ora` file.

Stop and Start the Oracle Net Listener for the Gateway

You must stop and restart the Oracle Net Listener to initiate the new settings, as follows:

1. Set the `PATH` environment variable to `$ORACLE_HOME/bin` where `$ORACLE_HOME` is the directory in which the gateway is installed.

For example on the Linux platform, if you have the Bourne or Korn Shell, enter the following:

```

$ PATH=$ORACLE_HOME/bin:$PATH;export PATH
$ LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH; export LD_LIBRARY_PATH

```

If you have the C Shell, enter the following:

```

$ setenv PATH $ORACLE_HOME/bin:$PATH
$ setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH

```

Table 11–1 specifies which parameter value to use for the different platforms:

Table 11–1 Parameter Values for UNIX Based Platforms

Platform	Parameter Value
Solaris (SPARC) 64 bit	<code>LD_LIBRARY_PATH=\$ORACLE_HOME/lib</code>
HP-UX PA-RISC	<code>SHLIB_PATH=\$ORACLE_HOME/lib</code>

Table 11–1 (Cont.) Parameter Values for UNIX Based Platforms

Platform	Parameter Value
HP-UX Itanium	LD_LIBRARY_PATH=\$ORACLE_HOME/lib
Linux x86, and Linux x86 64 bit	LD_LIBRARY_PATH=\$ORACLE_HOME/lib
AIX	LIBPATH=\$ORACLE_HOME/lib

- If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

- Check the status of the listener with the new settings, as follows:

```
$ lsnrctl status
```

The following is a partial output from a `lsnrctl` status check. In this example `dg4odbc` is the SID.

```
.
.
.
Services Summary...
Service "dg4odbc" has 1 instance(s).
  Instance "dg4odbc", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

Configure the Oracle Database for Gateway Access

Before you use the gateway to access an ODBC data source you must configure the Oracle database to enable communication with the gateway over Oracle Net.

To configure the Oracle database you must add connect descriptors to the `tnsnames.ora` file. By default, this file is in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory in which the Oracle database is installed. You cannot use the Oracle Net Assistant or the Oracle Net Easy Config tools to configure the `tnsnames.ora` file. You must edit the file manually.

A sample of the `tnsnames.ora` entry (`tnsnames.ora.sample`) is available in the `$ORACLE_HOME/dg4odbc/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

See Also: *Oracle Database Administrator's Guide* for information about editing the `tnsnames.ora` file.

Configuring `tnsnames.ora`

Edit the `tnsnames.ora` file to add a connect descriptor for the gateway. The following is a syntax of the Oracle Net entry using the TCP/IP protocol:

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name)
      (PORT=port_number)
    )
```



```
(CONNECT_DATA=
  (SID=gateway_sid)
(HS=OK))
```

Where:

Variable	Description
<i>connect_descriptor</i>	is the description of the object to connect to as specified when creating the database link, such as <code>dg4odbc</code> . Check the <code>sqlnet.ora</code> file for the following parameter setting: <code>names.directory_path = (TNSNAMES)</code> Note: The <code>sqlnet.ora</code> file is typically stored in <code>\$ORACLE_HOME/network/admin</code> .
TCP	is the TCP protocol used for TCP/IP connections.
<i>host_name</i>	specifies the machine where the gateway is running.
<i>port_number</i>	matches the port number used by the Oracle Net Listener that is listening for the gateway. The Oracle Net Listener's port number can be found in the <code>listener.ora</code> file used by the Oracle Net Listener. See " Syntax of listener.ora File Entries " on page 11-3.
<i>gateway_sid</i>	specifies the SID of the gateway and matches the SID specified in the <code>listener.ora</code> file of the Oracle Net Listener that is listening for the gateway. See " Configure Oracle Net Listener for the Gateway " on page 11-3 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

Create Database Links

Any Oracle client connected to the Oracle database can access an ODBC data source through the gateway. The Oracle client and the Oracle database can reside on different machines. The gateway accepts connections only from the Oracle database.

A connection to the gateway is established through a database link when it is first used in an Oracle session. In this context, a connection refers to the connection between the Oracle database and the gateway. The connection remains established until the Oracle session ends. Another session or user can access the same database link and get a distinct connection to the gateway and ODBC data source.

Database links are active for the duration of a gateway session. If you want to close a database link during a session, you can do so with the `ALTER SESSION` statement.

To access the ODBC data source, you must create a database link. A public database link is the most common of database links.

```
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO
2 "user" IDENTIFIED BY "password" USING 'tns_name_entry';
```

Where:

Variable	Description
<i>dblink</i>	is the complete database link name.
<i>tns_name_entry</i>	specifies the Oracle Net connect descriptor specified in the <code>tnsnames.ora</code> file that identifies the gateway

After the database link is created you can verify the connection to the ODBC data source, as follows:

```
SQL> SELECT * FROM DUAL@dblink;
```

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using database links.

Encrypt Gateway Initialization Parameter Values

The gateway uses user IDs and passwords to access the information in the remote database. Some user IDs and passwords must be defined in the gateway initialization file to handle functions such as resource recovery. In the current security conscious environment, having plain-text passwords that are accessible in the initialization file is deemed insecure. The `dg4pwd` encryption utility has been added as part of Heterogeneous Services to help make this more secure. This utility is accessible by this gateway. The initialization parameters which contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using this utility.

Configure the Gateway to Access Multiple ODBC Data Sources

The tasks for configuring the gateway to access multiple ODBC data sources are similar to the tasks for configuring the gateway for a single data source. The configuration example assumes the following:

- The gateway is installed and configured with the SID of `dg4odbc`.
- The gateway is configured to access one ODBC data source named `dsn1`.
- Two ODBC data sources named `dsn2` and `dsn3` where `dsn2` and `dsn3` are the data source names configured in the `odbc.ini` file, are being added.

Multiple ODBC Data Sources Example: Configuring the Gateway

Choose One System ID for Each ODBC Data Source

A separate instance of the gateway is needed for each ODBC data source. Each instance needs its own gateway System ID (SID). For this example, the gateway SIDs are chosen for the instances that access the ODBC data source:

- `dg4odbc2` for the gateway accessing data source `dsn2`.
- `dg4odbc3` for the gateway accessing data source `dsn3`.

Create Two Initialization Parameter Files

Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file `$ORACLE_HOME/hs/admin/initdg4odbc.ora`, twice, naming one with the gateway SID for `dsn2` and the other with the gateway SID for `dsn3`:

```
$ cd ORACLE_HOME/hs/admin
$ cp initdg4odbc.ora initdg4odbc2.ora
$ cp initdg4odbc.ora initdg4odbc3.ora
```

Change the value of the `HS_FDS_CONNECT_INFO` parameter in the new files, as follows:

For `initdg4odbc2.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=dsn2
```

For `initdg4odbc3.ora`, enter the following:

```
HS_FDS_CONNECT_INFO=dsn3
```

Note: If you have multiple gateway SIDs for the same ODBC data source because you want to use different gateway parameter settings at different times, follow the same procedure. You create several initialization parameter files, each with different SIDs and different parameter settings.

Multiple ODBC Data Sources Example: Configuring Oracle Net Listener

Add Entries to `listener.ora`

Add two new entries to the Oracle Net Listener configuration file, `listener.ora`. You must have an entry for each gateway instance, even when multiple gateway instances access the same database.

The following example shows the entry for the original installed gateway first, followed by the new entries. The syntax for HP-UX PA-RISC slightly different than the other platforms.

```
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4odbc)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4odbc)
    (ENVS=LD_LIBRARY_PATH=odbc_library_dir:$ORACLE_HOME/lib)
  )
  (SID_DESC=
    (SID_NAME=dg4odbc2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4odbc)
    (ENVS=LD_LIBRARY_PATH=odbc_library_dir:$ORACLE_HOME/lib)
  )
  (SID_DESC=
    (SID_NAME=dg4odbc3)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4odbc)
    (ENVS=LD_LIBRARY_PATH=odbc_library_dir:$ORACLE_HOME/lib)
  )
)
```

where, `oracle_home_directory` is the directory where the gateway resides.

Multiple ODBC Data Sources Example: Stopping and Starting the Oracle Net Listener

If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
```

```
$ lsnrctl start
```

Multiple ODBC Data Sources Example: Configuring Oracle Database for Gateway Access

Add two connect descriptor entries to the `tnsnames.ora` file. You must have an entry for each gateway instance, even if the gateway instances access the same database.

The following example shows the entry for the original installed gateway first, followed by the two entries for the new gateway instances:

```
old_dsn_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4odbc) )
  (HS=OK) )
new_dsn2_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4odbc2) )
  (HS=OK) )
new_dsn3_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4odbc3) )
  (HS=OK) )
```

The value for `PORT` is the TCP/IP port number of the Oracle Net Listener that is listening for the gateway. The number can be found in the `listener.ora` file used by the Oracle Net Listener. The value for `HOST` is the name of the machine on which the gateway is running. The name also can be found in the `listener.ora` file used by the Oracle Net Listener.

Multiple ODBC Data Sources Example: Accessing ODBC Data

Enter the following to create a database link for the `dg4odbc2` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK ODBC2 CONNECT TO
  2 "user2" IDENTIFIED BY "password2" USING 'new_dsn2_using';
```

Enter the following to create a database link for the `dg4odbc3` gateway:

```
SQL> CREATE PUBLIC DATABASE LINK ODBC3 CONNECT TO
  2 "user3" IDENTIFIED BY "password3" USING 'new_dsn3_using';
```

After the database links are created, you can verify the connection to the new ODBC data sources, as in the following:

```
SQL> SELECT * FROM ALL_USERS@ODBC2;
```

```
SQL> SELECT * FROM ALL_USERS@ODBC3;
```


Part VII

Installing and Configuring Oracle Database Gateway for DRDA

Part VII, "Installing and Configuring Oracle Database Gateway for DRDA" describes how to install and configure Oracle Database Gateway for DRDA on UNIX based platforms.

It contains the following chapters:

- Chapter 12, "Installing Oracle Database Gateway for DRDA"
- Chapter 13, "Configuring the DRDA Server"
- Chapter 14, "Configuring Oracle Database Gateway for DRDA"
- Chapter 15, "Security Considerations"
- Chapter 16, "Migration From Previous Releases"

Installing Oracle Database Gateway for DRDA

This chapter provides information about the hardware and software requirements and the installation procedure of Oracle Database Gateway for DRDA.

To install the gateway, follow these steps:

1. Ensure that the system meets all of the hardware and software requirements specified in "[System Requirements for Oracle Database Gateway for DRDA](#)" on page 12-1.
2. Run the Oracle Universal Installer.

See "[Step through the Oracle Universal Installer](#)" on page 12-4 for more information about running the Oracle Universal Installer.

Oracle Universal Installer is a menu-driven utility that guides you through the installation of the gateway by prompting you with action items. The action items and the sequence in which they appear depend on your platform.

See [Table 12-3](#) for a description of the installation procedure of Oracle Database Gateway for DRDA.

System Requirements for Oracle Database Gateway for DRDA

This section provides information about the hardware and software requirements for the gateway. It contains the following sections:

- "[Hardware Requirements](#)" on page 12-1
- "[Software Requirements](#)" on page 12-3

Hardware Requirements

[Table 12-1](#) shows the minimum hardware requirements for Oracle Database Gateway for DRDA.

Table 12-1 Hardware requirements for Oracle Database Gateway for DRDA

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Temporary Disk Space	400 MB	400 MB	400 MB	400 MB	400 MB
Disk Space	1.5 GB	1.5 GB	1.5 GB	1.5 GB	1.5 GB

Table 12–1 (Cont.) Hardware requirements for Oracle Database Gateway for DRDA

Hardware Items	Required for AIX-Based System	Required for HP 9000 Series HP-UX PA-RISC	Required for Solaris Operating System (SPARC)	Required for Linux x86	Required for Linux x86 64 bit
Physical Memory*	256 MB	256 MB	256 RAM	256 RAM	256 RAM
Swap Space	1 GB	1 GB	1 GB	1 GB	1 GB
Processor	IBM pSeries	HP 9000 Series HP-UX that can run the required version of HP-UX	A Solaris Operating System (SPARC 64-bit) that can run the required version of Solaris with 64-bit architecture	x86	x86_64

* The minimum swap space is 1 GB (or twice the size of RAM). On systems with 2 GB or more of RAM, the swap space can be between one and two times the size of RAM. On AIX systems with 1 GB or more of memory, do not increase the swap space more than 2 GB.

For most installations, a minimum of 256 MB of real memory is recommended for the first user to support the Oracle Database Gateway for DRDA.

The total real memory requirement for each concurrent use of the gateway depends on the following factors:

- Number of concurrent TCP/IP connections open by each user
- Number of data items being transferred between the gateway and the remote transaction program
- Additional factors such as configured network buffer size

Checking the Hardware Requirements

To ensure that the system meets the minimum requirements, follow these steps:

1. To determine the physical RAM size, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lstat -E -l sys0 -a realmem
HP-UX PA-RISC	# /usr/sbin/dmesg grep "Physical:"
Solaris (SPARC)	# /usr/sbin/prtconf grep "Memory size"
Linux x86	# grep MemTotal /proc/meminfo
Linux x86 64 bit	# grep MemTotal /proc/meminfo

If the size of the physical RAM installed in the system is less than the required size, you must install more memory before continuing.

2. To determine the size of the configured swap space, enter one of the following commands:

Operating System	Command
AIX	# /usr/sbin/lspv -a
HP-UX PA-RISC	# /usr/sbin/swapinfo -a
Solaris (SPARC)	# /usr/sbin/swap -s

Operating System	Command
Linux x86	# grep SwapTotal /proc/meminfo
Linux x86 64 bit	# grep SwapTotal /proc/meminfo

If necessary, see your operating system documentation for information about how to configure additional swap space.

- To determine the amount of disk space available in the /tmp directory enter the following commands:

Operating System	Command
AIX	# df -k /tmp
HP-UX PA-RISC	# df -k /tmp
Solaris (SPARC)	# df -k /tmp
Linux x86	# df -k /tmp
Linux x86 64 bit	# df -k /tmp

- To determine the amount of disk space available on the system enter the following commands:

Operating System	Command
AIX	# df -k
HP-UX PA-RISC	# df -k
Solaris (SPARC)	# df -k
Linux x86	# df -k
Linux x86 64 bit	# df -k

Software Requirements

The following section describes the minimum software requirements for Oracle Database Gateway for DRDA.

Operating System

Table 12–2 shows the minimum operating system version required for Oracle Database Gateway for DRDA. If your operating system is lower than the minimum requirements, upgrade your operating system to meet the specified levels.

Table 12–2 Operating Systems version for Oracle Database Gateway for DRDA

Operating System	Version
AIX	AIX 5L version 5.3, Maintenance level 02 or higher
HP-UX PA-RISC	HP-UX 11i v2 (11.23) or HP-UX 11i v3 (11.31)
Solaris (SPARC)	Solaris 9 Update 6 or higher or Solaris 10
Linux x86 Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0

Table 12–2 (Cont.) Operating Systems version for Oracle Database Gateway for DRDA

Operating System	Version
Linux x86 Suse	SUSE Linux Enterprise Server 10.0
Linux x86 64 bit Red Hat	One of the following operating system versions: <ul style="list-style-type: none"> ■ Red Hat Enterprise Linux 4.0 ■ Red Hat Enterprise Linux 5.0
Linux x86 64 bit Suse	SUSE Linux Enterprise Server 10.0
Oracle Enterprise Linux x86	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0
Oracle Enterprise Linux x86 64 bit	One of the following operating system versions: <ul style="list-style-type: none"> ■ Oracle Enterprise Linux 4.0 ■ Oracle Enterprise Linux 5.0

Checking the Software Requirements

To ensure that the system meets the minimum requirements, follow these steps:

- To determine which version of AIX is installed, enter the following command:

```
# oslevel -r
```
- To determine which version of HP-UX PA-RISC is installed, enter the following command:

```
# uname -a
```
- To determine which version of Solaris Operating System (SPARC) is installed, enter the following command:

```
# uname -r
```
- To determine which distribution and version of Linux x86 is installed, enter the following command:

```
# cat /etc/issue
```
- To determine which distribution and version of Linux x86 64 bit is installed, enter the following command:

```
# cat /proc/version
```

Certified Configuration

The gateway supports DB2/UBD, DB2/OS390, and DB2/400. For the latest versions supported refer to the OTN Web site:

http://otn.oracle.com/products/gateways/pdf/mainframe_certification.pdf

Step through the Oracle Universal Installer

Start the Installer with the following command:

```
$ ./runInstaller
```

Table 12–3 describes the installation procedure for Oracle Database Gateway for DRDA.

Table 12–3 The Oracle Universal Installer: Steps for Installing the Gateway

Screen	Response
Oracle Universal Installer: Welcome	Click Next .
Oracle Universal Installer: Specify Home Details	Specify a name for the installation in the Name field. You can also choose not to edit the default setting of the Name field of the Specify Home Details screen. The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to ORACLE_HOME. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen.
Oracle Universal Installer: Available Product Components	a. Select Oracle Database Gateway for DRDA 11.1.0.6.0. b. Click Next.
Oracle Universal Installer: Summary	The Installation Summary screen enables you to review a tree list of options and components for this installation. Click Install to start installation.
Oracle Net Configuration Assistant: Welcome	Click Cancel .
Oracle Net Configuration Assistant:	Click Yes .
Oracle Universal Installer: Configuration Tools	Click Exit .
Exit	The final screen of the Oracle Universal Installer is the End of Installation screen. Click Exit to exit the installer.

Configuring the DRDA Server

This chapter describes tasks you must perform to configure the DRDA server. Each supported operating system is addressed separately. Experience with the given operating system and database is required.

The steps for configuring your remote DRDA server apply to the following DRDA servers:

- DB2/OS390
- DB2/400
- DB2/UDB (Universal Database)

Configuring a DRDA database to enable access by the gateway requires actions on the DRDA database and on certain components of the host operating system. Although no Oracle software is installed on the host system, access to and some knowledge of the host system and DRDA database are required during the configuration. Refer to the vendor documentation for complete information about your host system and DRDA database.

This chapter contains the following sections:

- ["Configuring the DRDA Server for DB2/OS390"](#) on page 13-1
- ["Configuring the DRDA Server for DB2/400"](#) on page 13-3
- ["Configuring the DRDA Server for DB2/UDB \(Universal Database\)"](#) on page 13-3

Configuring the DRDA Server for DB2/OS390

Perform the following tasks to configure the DRDA server with DB2 on an OS390 system:

1. Define the user ID that owns the package

During gateway configuration, you will need to run the Bind Package Stored Procedure to bind the gateway package on the DRDA Server. To properly bind the package, the user ID and password that are used when the procedure is run (either implied as the current Oracle user or explicitly defined in the `CREATE DATABASE LINK` command) must have proper authority on the DRDA Server to create the package. This user ID should be used to create and own the `ORACLE2PC` (two-phase commit) table. The user ID that is used to bind or rebind the DRDA package must have one or more of the following privileges on the DRDA Server:

- Package privileges of `BIND`, `COPY`, and `EXECUTE`, for example:

```
GRANT BIND    ON PACKAGE drda1.* TO userid
GRANT COPY    ON PACKAGE drda1.* TO userid
```

```
GRANT EXECUTE ON PACKAGE drda1.* TO PUBLIC
```

- Collection privilege of CREATE IN, for example:

```
GRANT CREATE IN ON PACKAGE drda1 TO USER userid
```

- System privileges of BINDADD and BINDAGENT, for example:

```
GRANT BINDADD TO USER userid  
GRANT BINDAGENT TO USER userid
```

- Database privilege of CREATETAB, for example:

```
GRANT CREATETAB ON DATABASE database TO USER userid
```

Choose a user ID that will own the package and the ORACLE2PC table. Ensure that this user ID is defined to both DB2 and OS/390 (MVS).

2. Define the recovery user ID

During gateway configuration, the recovery user ID and password are specified in the Gateway Initialization File using the `DRDA_RECOVERY_USERID` and `DRDA_RECOVERY_PASSWORD` parameters. If a distributed transaction fails, then the recovery process connects to the remote database using the user ID and password defined in these parameters. This user ID must have execute privileges on the package and must be defined in the DRDA database. If the user ID is not specified in `DRDA_RECOVER_USERID`, then the gateway attempts to connect to a user ID of `ORARECOV` when a distributed transaction is in doubt.

Determine the user ID and password you will use for recovery.

3. Determine DRDA location name for DB2 instance

The DRDA location name is required as a gateway parameter. To determine the location name, run the following SQL query from a DB2 SPUFI session:

```
SELECT CURRENT SERVER FROM any_table
```

where `any_table` is a valid table with one or more rows.

If the value returned by this query is blank or null, then the DRDA location name has not been established. Contact the system administrator to arrange to set a location name for the instance.

4. Configure DB2 Distributed Data Facility for Gateway

DB2 Distributed Data Facility (DDF) is the component of DB2 that manages all distributed database operations, both DRDA and non-DRDA.

If your site uses DB2 distributed operations, then DDF is probably operational on the DB2 instance you plan to access through the gateway. If DDF is not operational, then you must configure it and start it as described in the appropriate DB2 documentation.

Even if DDF is operational on the DB2 instance, it might be necessary to make changes to the DDF Communication Database (CDB) tables to specify the authorization conduct of DRDA sessions from the gateway. This can be done by properly authorized users with a utility like the DB2 SPUFI utility. If you make changes to CDB tables, then you must stop and restart DDF for the changes to take effect. Refer to Chapter 14, "Security Considerations", for additional CDB tables and security information.

Configuring the DRDA Server for DB2/400

Experience with DB2/400 and AS/400 is required to perform the following steps:

1. Define the user ID that owns the package

During gateway configuration, you will need to run the Bind Package Stored Procedure to bind the gateway package on the DRDA Server. To properly bind the package, the user ID and password used when the procedure is run (either implied as the current Oracle user or explicitly defined in the `CREATE DATABASE LINK` command) must have proper authority on the DRDA Server to create the package. This user ID should be used to create and own the `ORACLE2PC` (two-phase commit) table. The user ID that is used to bind or rebind the DRDA package must have the following privileges on the DRDA Server:

- Use authority on the `CRTSQPKG` command
- Change authority on the library the package will be created in

Choose a user ID now that will own the package and `ORACLE2PC` table. Ensure that this user ID is defined in DB2/400 and AS/400.

2. Define the recovery user ID

During gateway configuration, the recovery user ID and password are specified in the Gateway Initialization File using the `DRDA_RECOVERY_USERID` and `DRDA_RECOVERY_PASSWORD` parameters. If a distributed transaction fails, then the recovery process connects to the remote database using the user ID and password defined in these parameters. This user ID must have execute privileges on the package and must be defined to the DRDA database. If the user ID is not specified in `DRDA_RECOVER_USERID`, then the gateway attempts to connect to a user ID of `ORARECOV` when a distributed transaction is in doubt.

Determine the user ID and password you will use for recovery.

3. Determine DRDA location name for DB2/400 instance

The DRDA location name is required as a gateway parameter. To determine the location name, run the following SQL query from a STRSQL session. If SQL is unavailable on the system, then use the AS/400 command `DSPRDBDIRE` to identify your LOCAL DRDA Server.

```
SELECT CURRENT SERVER FROM any_table
```

where *any_table* is a valid table with one or more rows.

If the value returned by this query is blank or null, then the DRDA location name has not been established. Contact the system administrator to arrange to set a location name for the instance.

Configuring the DRDA Server for DB2/UDB (Universal Database)

Experience with DB2/UDB, configuring the communication subsystem of DB2/UDB, and the host System Administration tools is required to perform the following steps:

1. Define the user ID that owns the package

During gateway configuration, you will need to run the Bind Package Stored Procedure to bind the gateway package on the DRDA Server. To properly bind the package, the user ID and password used when the procedure is run (either implied as the current Oracle user or explicitly defined in the `CREATE DATABASE LINK` command) must have proper authority on the DRDA Server to create the

package. This user ID should be used to create and own the ORACLE2PC (two-phase commit) table. The user ID that is used to bind or rebind the DRDA package must have one or more of the following privileges on the DRDA Server:

- Package privileges of BIND and EXECUTE, for example:

```
GRANT BIND    ON PACKAGE drda1.g2drsqli TO USER userid
GRANT EXECUTE ON PACKAGE drda1.g2drsqli TO PUBLIC
```

- Schema privileges of CREATEIN, for example:

```
GRANT CREATEIN ON SCHEMA otgdb2 TO USER userid
GRANT CREATEIN ON SCHEMA drda1 TO USER userid
```

- Database authorities of CONNECT, BINDADD, and CREATETAB, for example:

```
GRANT CONNECT  ON DATABASE TO USER userid
GRANT BINDADD  ON DATABASE TO USER userid
GRANT CREATETAB ON DATABASE TO USER userid
```

Choose a user ID now that will own the package and ORACLE2PC table. Ensure that this user ID is defined in both the DB2 instance ID and the operating system

2. Define the recovery user ID

During gateway configuration, the recovery user ID and password are specified in the Gateway Initialization File using the DRDA_RECOVERY_USERID and DRDA_RECOVERY_PASSWORD parameters. If a distributed transaction fails, then the recovery process connects to the remote database using the user ID and password defined in these parameters. This user ID must have execute privileges on the package and must be defined to the DRDA database. If the user ID is not specified in DRDA_RECOVER_USERID, then the gateway attempts to connect to a user ID of ORARECOV when a distributed transaction is in doubt.

Determine the user ID and password you will use for recovery.

3. Determine DRDA location name for DB2/UDB instance

The DRDA location name is required as a gateway parameter. To determine the location name, run the following SQL query from a DB2 CLI session:

```
SELECT CURRENT SERVER FROM any_table
```

where *any_table* is a valid table with one or more rows.

If the value returned by this query is blank or null, then the DRDA location name has not been established. Contact your system administrator to set a location name for the instance.

Configuring Oracle Database Gateway for DRDA

After installing the gateway, perform the following tasks to configure Oracle Database Gateway for DRDA:

1. [Configure the Gateway Initialization Parameter File](#)
2. [Configure Oracle Net for the Gateway](#)
3. [Configure Two-Phase Commit](#)
4. [Bind the DRDA Gateway Package](#)
5. [Create Tables and Views for Data Dictionary Support](#)
6. [Grant Authority to the DRDA Package](#)
7. [Configure the Oracle Database for Gateway Access](#)
8. [Create Database Links](#)
9. [Configure the Gateway to Access Multiple DRDA Databases](#)

SQL scripts are provided to perform steps such as creating the ORACLE2PC table, removing obsolete tables and views, and creating tables and views to provide data dictionary support.

These scripts must be run on the DRDA Server platform using a database native tool (such as SPUFI on DB2/OS390), because no tool is provided with the gateway to execute these scripts. Note that when running these scripts, the user ID used must be suitably authorized.

Choose the appropriate subdirectory for your DRDA Server platform from the following list:

- for DB2/OS390: choose dg4drda/install/db2
- for DB2/400: choose dg4drda/install/as400
- for DB2/UDB: choose dg4drda/install/db2udb

Configure the Gateway Initialization Parameter File

Perform the following tasks to configure the gateway initialization parameter file:

1. [Choose a System Identifier for the Gateway](#)
2. [Customize the Initialization Parameter File](#)

Choose a System Identifier for the Gateway

The gateway system identifier (SID) is an alphanumeric character string that identifies a gateway instance. You need one gateway instance, and therefore one gateway SID, for each DRDA database you are accessing. However, if you want to access two DRDA databases, you need two gateway SIDs, one for each instance of the gateway. If you have one DRDA database and want to access it sometimes with one set of gateway parameter settings, and other times with different gateway parameter settings, you can do that by having multiple gateway SIDs for the single DRDA database. The SID is used as part of the file name for the initialization parameter file.

Customize the Initialization Parameter File

To configure the host for the Oracle Database Gateway for DRDA, you must tailor the parameter files for your installation. Perform the following steps:

1. Choose the `initsid.ora` file

The `initsid.ora` gateway initialization file defines the operating parameters for the gateway. Samples (tailored for each type of DRDA Server) are provided as a starting point for tailoring to your particular installation. The samples are stored in the `$ORACLE_HOME/dg4drda/admin` directory. The following is a list of the initialization files for various DRDA Server platforms:

- For DB2/OS390: `initDB2.ora`
- For DB2/UDB: `initDB2UDB.ora`
- For DB2/400: `initAS400.ora`

Choose a sample initialization file and copy it, within the same directory, using the following naming convention:

```
initsid.ora
```

where `sid` is the chosen gateway SID. For example, if the chosen gateway SID were DRDA, then the initialization file would be named `initDRDA.ora`.

2. Tailor the `initsid.ora` file

After you have copied the sample initialization file, you will need to tailor it to your installation. While many parameters can be left to their defaults, some parameters must be changed for correct operation of the gateway. Attention should be given to the following DRDA and HS parameters. Attention should also be given to the security aspects of the initialization file. [Chapter 15, "Security Considerations"](#), contains details concerning encryption of passwords that would otherwise be embedded in the initialization file. See [Appendix C, "Initialization Parameters"](#), for a description of the following parameters:

- `DRDA_CONNECT_PARM`
- `DRDA_PACKAGE_COLLID`
- `DRDA_PACKAGE_NAME`
- `DRDA_PACKAGE_OWNER`
- `DRDA_REMOTE_DB_NAME`
- `HS_DB_NAME`
- `HS_DB_DOMAIN`
- `FDS_CLASS`

Configure Oracle Net for the Gateway

The gateway requires Oracle Net to communicate with the Oracle database. After configuring the gateway, perform the following tasks to configure Oracle Net to work with the gateway:

1. [Configure Oracle Net Listener for the Gateway](#)
2. [Stop and Start the Oracle Net Listener for the Gateway](#)

Configure Oracle Net Listener for the Gateway

The Oracle Net Listener listens for incoming requests from the Oracle database. For the Oracle Net Listener to listen for the gateway, information about the gateway must be added to the Oracle Net Listener configuration file, `listener.ora`. This file by default is located in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory under which the gateway is installed.

The following entries must be added to the `listener.ora` file:

- A list of Oracle Net addresses on which the Oracle Net Listener listens
- The executable name of the gateway that the Oracle Net Listener starts in response to incoming connection requests

A sample of the `listener.ora` entry (`listener.ora.sample`) is available in the `$ORACLE_HOME/dg4drda/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

Syntax of listener.ora File Entries

The Oracle database communicates with the gateway using Oracle Net and any supported protocol adapters. The following syntax of the address on which the Oracle Net Listener listens using the TCP/IP protocol adapter:

```
LISTENER=
  (ADDRESS=
    (PROTOCOL=TCP)
    (HOST=host_name)
    (PORT=port_number))
```

Where:

Variable	Description
<i>host_name</i>	is the name of the machine on which the gateway is installed.
<i>port_number</i>	specifies the port number used by the Oracle Net Listener. If you have other listeners running on the same machine, then the value of <i>port_number</i> must be different from the other listeners' port numbers.

To direct the Oracle Net Listener to start the gateway in response to incoming connection requests, add an entry to the `listener.ora` file.

Note: You must use the same SID value in the `listener.ora` file and the `tnsnames.ora` file which will be configured in the next step.

```
SID_LIST_LISTENER=
  (SID_LIST=
```

```

        (SID_DESC=
          (SID_NAME=gateway_sid)
          (ORACLE_HOME=oracle_home_directory)
          (PROGRAM=g4drsrv)
        )
      )

```

Where:

Variable	Description
<i>gateway_sid</i>	specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <code>tnsnames.ora</code> file.
<i>oracle_home_directory</i>	specifies the Oracle home directory where the gateway resides.
<i>g4drsrv</i>	specifies the executable name of the Oracle Database Gateway for DRDA.

If you are already running a Oracle Net Listener that listens on multiple database SIDs, add only the following syntax to `SID_LIST` in the existing `listener.ora` file:

```

SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=.
    .
  )
  (SID_DESC=.
    .
  )
  (SID_DESC=
    (SID_NAME=gateway_sid)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=g4drsrv)
  )
)

```

See Also: *Oracle Database Net Services Administrator's Guide Administrator's Guide* for information about changing the `listener.ora` file.

Stop and Start the Oracle Net Listener for the Gateway

You must stop and restart the Oracle Net Listener to initiate the new settings, as follows:

1. Set the `PATH` environment variable to `$ORACLE_HOME/bin` where `$ORACLE_HOME` is the directory in which the gateway is installed. If you have the Bourne or Korn Shell, enter the following:

```

$ PATH=$ORACLE_HOME/bin:$PATH;export PATH
$ LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH; export LD_LIBRARY_PATH

```

If you have the C Shell, enter the following:

```

$ setenv PATH $ORACLE_HOME/bin:$PATH
$ setenv LD_LIBRARY_PATH $ORACLE_HOME/lib:$LD_LIBRARY_PATH

```

[Table 14-1](#) specifies which parameter value to use for the different platforms:

Table 14–1 Parameter Values for UNIX Based Platforms

Platform	Parameter Value
Solaris (SPARC) 64 bit	LD_LIBRARY_PATH=\$ORACLE_HOME/lib
HP-UX PA-RISC	SHLIB_PATH=\$ORACLE_HOME/lib
Linux x86, and Linux x86 64 bit	LD_LIBRARY_PATH=\$ORACLE_HOME/lib
AIX	LIBPATH=\$ORACLE_HOME/lib

2. If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

3. Check the status of the listener with the new settings, as follows:

```
$ lsnrctl status
```

The following is a partial output from a `lsnrctl` status check:

```
.
.
.
Listening Endpoints Summary...
  (DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=204.179.99.15) (PORT=1551)))
Services Summary...
Service "dg4drda" has 1 instance(s).
  Instance "dg4drda", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

In this example, the service name is `dg4drda` which is the default SID value assigned during installation.

Note: You must use the same SID value in the `tnsnames.ora` file and the `listener.ora` file.

Configure Two-Phase Commit

Run the appropriate script depending on the server you are connecting to:

- If connecting to DB2/UDB, then perform the following steps on the DRDA server platform, to create the `ORACLE2PC` table:
 - a. Log into the machine where DB2/UDB is running.

Check that you have the ability to address the DB2/UDB instance where the `ORACLE2PC` table will reside.
 - b. Copy the files from the `$ORACLE_HOME/dg4drda/install/db2udb` directory.
 - `o2pc.sh` (Sample shell script for performing the table creation)
 - `o2pc.sql` (SQL script for creating the table)
 - `o2pcg.sql` (SQL script for granting package access to PUBLIC)
 - c. Connect to the database using the user ID that you will use for binding the package:

```
$ db2 'CONNECT TO database USER userid USING password'
```

Note: The user ID must have CONNECT, CREATETAB, and BINDADD authority to be able to connect to the database, create the table, and create the package.

For more information, refer to "[Configuring the DRDA Server for DB2/UDB \(Universal Database\)](#)" on page 13-3.

- d. Create the ORACLE2PC table:

```
$ db2 -tf o2pc.sql
```

- e. Commit the transaction:

```
$ db2 'COMMIT'
```

- f. Verify that the table was created.

Optionally, verify the table was created under the correct user ID:

```
$ db2 'LIST TABLES FOR USER'
```

```
$ db2 'COMMIT'
```

- g. Disconnect from the session:

```
$ db2 'DISCONNECT CURRENT'
```

- If connecting to DB2/400, then run the following SQL script on the DRDA server platform using a database native tool, to create your ORACLE2PC table:

```
$ORACLE_HOME/dg4drda/install/as400/o2pc.sql
```

- If connecting to DB2/OS390, then run the following SQL script on the DRDA server platform using a database native tool, to create your ORACLE2PC table:

```
$ORACLE_HOME/dg4drda/install/db2/o2pc.sql
```

Bind the DRDA Gateway Package

The product requires a package to be bound on the DRDA Server. The gateway has an internal, stored procedure that must be used to create this package. The internal, stored procedure is invoked from an Oracle database. Before this package can be bound on the DRDA server, the gateway initialization file must be correctly configured. Refer to [Appendix C, "Initialization Parameters"](#) for more information.

DRDA Gateway Package Binding Considerations

Before binding the DRDA gateway package, perform the following steps:

1. Check the user has the appropriate privileges

The DRDA package must be bound with the internal stored procedure GTW\$_BIND_PKG. The user ID used to bind the DRDA package must have the appropriate privileges on the remote database, as described in [Chapter 13, "Configuring the DRDA Server"](#).

2. Check all DRDA parameter settings

Check all DRDA parameter settings to be sure that they are set correctly before you start the bind. For example, the default for DRDA_DISABLE_CALL only works if your DRDA database supports stored procedures. If not, then you must change

the setting. Also, the value for `DRDA_PACKAGE_NAME` must be unique if you have any older versions of the gateway installed. New packages replace any old packages with the same name, causing versions of the gateway that use the old package to fail. Refer to [Appendix C, "Initialization Parameters"](#) for information on the parameters and their settings.

DRDA Gateway Package Binding Steps

Perform the following steps:

1. Log on to an Oracle database.

Use SQL*Plus:

```
$ sqlplus system/manager
```

2. Create a database link using the user ID that you chose while configuring the DRDA Server.

```
SQL> CREATE PUBLIC DATABASE LINK dblink
2 CONNECT TO userid IDENTIFIED BY password
3 USING 'tns_name_entry'
```

Note: The user ID that is creating the public database link must have the `CREATE PUBLIC DATABASE LINK` privilege.

3. Execute the stored procedure `GTW$_BIND_PKG`:

```
SQL> exec GTW$_BIND_PKG@dblink;
SQL> COMMIT;
```

This creates and commits the package. If any errors are reported, then correct the Gateway Initialization File parameters as needed and re-execute the bind procedure above.

Create Tables and Views for Data Dictionary Support

To enable data dictionary translation support, data dictionary tables and views have to be created on each non-Oracle system that you want to access through the gateway.

Perform the following steps to create the data dictionary tables and views using database native tools:

1. Upgrade from a previous gateway release

If you are upgrading from a previous version of the gateway then run the appropriate script to drop the old data dictionary definitions.

- If connecting to DB2/UDB, then run


```
$ORACLE_HOME/dg4drda/install/db2udb/dropold.sql
```
- If connecting to DB2/OS390, then run


```
$ORACLE_HOME/dg4drda/install/db2/dropold.sql
```
- If connecting to DB2/400, then run


```
$ORACLE_HOME/dg4drda/install/as400/dropold.sql
```

2. Create the data dictionary tables

Run the appropriate script to create the data dictionary tables.

- If connecting to DB2/UDB, then run
`$ORACLE_HOME/dg4drda/install/db2udb/g4ddtab.sql`
- If connecting to DB2/OS390, then run
`$ORACLE_HOME/dg4drda/install/db2/g4ddtab.sql`
- If connecting to DB2/400, then run
`$ORACLE_HOME/dg4drda/install/as400/g4ddtab.sql`

3. Creating the data dictionary views

Run the appropriate script to create the data dictionary views.

- If connecting to DB2/UDB, then run
For DB2/UDB V7:
`$ORACLE_HOME/dg4drda/install/db2udb/g4ddvwu7.sql`
For DB2/UDB V8:
`$ORACLE_HOME/dg4drda/install/db2udb/g4ddvwu8.sql`
- If connecting to DB2/OS390 then run
For DB2/OS390 V7 (RACF security):
`$ORACLE_HOME/dg4drda/install/db2/g4ddvwr7.sql`
For DB2/OS390 V7 (DB2 security):
`$ORACLE_HOME/dg4drda/install/db2/g4ddvws7.sql`
For DB2/OS390 V8 (RACF security):
`$ORACLE_HOME/dg4drda/install/db2/g4ddvwr8.sql`
For DB2/OS390 V8 (DB2 security):
`$ORACLE_HOME/dg4drda/install/db2/g4ddvws8.sql`
- If connecting to DB2/400, then run
For DB2/400 V5.1:
`$ORACLE_HOME/dg4drda/install/as400/g4ddvw51.sql`
For DB2/400 V5.2:
`$ORACLE_HOME/dg4drda/install/as400/g4ddvw52.sql`

Grant Authority to the DRDA Package

To grant authority to the package, run the appropriate script on the non-oracle system:

- If connecting to DB2/UDB, then run
`$ORACLE_HOME/dg4drda/install/db2udb/o2pcg.sql`
- If connecting to DB2/OS390, then run
`$ORACLE_HOME/dg4drda/install/db2/o2pcg.sql`

- If connecting to DB2/400, then run

```
$ORACLE_HOME/dg4drda/install/as400/o2pcg.sql
```

Configure the Oracle Database for Gateway Access

Before you use the gateway to access DB2 data you must configure the Oracle database to enable communication with the gateway over Oracle Net.

To configure the Oracle database you must add connect descriptors to the `tnsnames.ora` file. By default, this file is in `$ORACLE_HOME/network/admin`, where `$ORACLE_HOME` is the directory in which the Oracle database is installed. You cannot use the Oracle Net Assistant or the Oracle Net Easy Config tools to configure the `tnsnames.ora` file. You must edit the file manually.

A sample of the `listener.ora` entry (`listener.ora.sample`) is available in the `$ORACLE_HOME/dg4drda/admin` directory where `$ORACLE_HOME` is the directory under which the gateway is installed.

See Also: *Oracle Database Administrator's Guide* for information about editing the `tnsnames.ora` file.

Configuring tnsnames.ora

Edit the `tnsnames.ora` file to add a connect descriptor for the gateway. The following is a syntax of the Oracle Net entry using the TCP/IP protocol:

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name)
      (PORT=port_number)
    )
    (CONNECT_DATA=
      (SID=gateway_sid)
    )
    (HS=OK) )
```

Where:

Variable	Description
<code>connect_descriptor</code>	is the description of the object to connect to as specified when creating the database link, such as <code>dg4drda</code> . Check the <code>sqlnet.ora</code> file for the following parameter setting: <ul style="list-style-type: none"> ■ <code>names.directory_path = (TNSNAMES)</code> Note: The <code>sqlnet.ora</code> file is typically stored in <code>\$ORACLE_HOME/network/admin</code> .
TCP	is the TCP protocol used for TCP/IP connections.
<code>host_name</code>	specifies the machine where the gateway is running.
<code>port_number</code>	matches the port number used by the Oracle Net Listener that is listening for the gateway. The Oracle Net Listener's port number can be found in the <code>listener.ora</code> file used by the Oracle Net Listener. See " Syntax of listener.ora File Entries " on page 14-3.

Variable	Description
<i>gateway_sid</i>	specifies the SID of the gateway and matches the SID specified in the <code>listener.ora</code> file of the Oracle Net Listener that is listening for the gateway. See "Configure Oracle Net Listener for the Gateway" on page 14-3 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

Create Database Links

Any Oracle client connected to the Oracle database can access DB2 data through the gateway. The Oracle client and the Oracle database can reside on different machines. The gateway accepts connections only from the Oracle database.

A connection to the gateway is established through a database link when it is first used in an Oracle session. In this context, a connection refers to the connection between the Oracle database and the gateway. The connection remains established until the Oracle session ends. Another session or user can access the same database link and get a distinct connection to the gateway and DRDA database.

Database links are active for the duration of a gateway session. If you want to close a database link during a session, you can do so with the `ALTER SESSION` statement.

To access the DRDA server, you must create a database link. A public database link is the most common of database links.

```
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO
2 "user" IDENTIFIED BY "password" USING 'tns_name_entry';
```

Where:

Variable	Description
<i>dblink</i>	is the complete database link name.
<i>tns_name_entry</i>	specifies the Oracle Net connect descriptor specified in the <code>tnsnames.ora</code> file that identifies the gateway

After the database link is created you can verify the connection to the DRDA database, as follows:

```
SQL> SELECT * FROM DUAL@dblink;
```

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity Administrator's Guide* for more information about using database links.

Configure the Gateway to Access Multiple DRDA Databases

The tasks for configuring the gateway to access multiple DRDA databases are similar to the tasks for configuring the gateway for a single database. The configuration example assumes the following:

- The gateway is installed and configured with the default SID of `dg4drda`
- The `ORACLE_HOME` environment variable is set to the directory where the gateway is installed

- The gateway is configured for one DRDA database named `db1`
- Two DRDA databases named `db2` and `db3` on a host with IP Address 204.179.79.15 are being added

Multiple DRDA Databases Example: Configuring the Gateway

Choose One System ID for Each DRDA Database

A separate instance of the gateway is needed for each DRDA database. Each instance needs its own gateway System ID (SID). For this example, the gateway SIDs are chosen for the instances that access the DRDA databases:

- `dg4drda2` for the gateway accessing database `db2`
- `dg4drda3` for the gateway accessing database `db3`

Create Two Initialization Parameter Files

Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file, `$ORACLE_HOME/dg4drda/admin/initdg4drda.ora`, twice, naming one with the gateway SID for `db2` and the other with the gateway SID for `db3`:

```
$ cd $ORACLE_HOME/dg4drda/admin
$ cp initdg4drda.ora initdg4drda2.ora
$ cp initdg4drda.ora initdg4drda3.ora
```

Note: If you have multiple gateway SIDs for the same DRDA database because you want to use different gateway parameter settings at different times, follow the same procedure. You create several initialization parameter files, each with different SIDs and different parameter settings.

Multiple DRDA Databases Example: Configuring Oracle Net Listener

Add Entries to listener.ora

Add two new entries to the Oracle Net Listener configuration file, `listener.ora`. You must have an entry for each gateway instance, even when multiple gateway instances access the same database.

The following example shows the entry for the original installed gateway first, followed by the new entries:

```
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4drda)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=g4drsrv)
  )
  (SID_DESC=
    (SID_NAME=dg4drda2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=g4drsrv)
  )
  (SID_DESC=
    (SID_NAME=dg4drda3)
```

```
(ORACLE_HOME=oracle_home_directory)
(PROGRAM=g4drsrv)
)
)
```

where, *oracle_home_directory* is the directory where the gateway resides.

Multiple DRDA Databases Example: Stopping and Starting the Oracle Net Listener

If the listener is already running, use the `lsnrctl` command to stop the listener and then start it with the new settings, as follows:

```
$ lsnrctl stop
$ lsnrctl start
```

Multiple Databases Example: Configuring Oracle Database for Gateway Access

Configuring Oracle Net for Multiple Gateway Instances

Add two connect descriptor entries to the `tnsnames.ora` file. You must have an entry for each gateway instance, even if the gateway instances access the same database.

The following DRDA example shows the entry for the original installed gateway first, followed by the two entries for the new gateway instances:

```
old_db_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4drda) )
  (HS=OK) )
new_db2_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4drda2) )
  (HS=OK) )
new_db3_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4drda3) )
  (HS=OK) )
```

The value for `PORT` is the TCP/IP port number of the Oracle Net Listener that is listening for the gateway. The number can be found in the `listener.ora` file used by the Oracle Net Listener. The value for `HOST` is the name of the machine on which the gateway is running. The name also can be found in the `listener.ora` file used by the Oracle Net Listener.

Multiple DRDA Databases Example: Accessing DB2 Data

Enter the following to create a database link for the dg4drda2 gateway:

```
SQL> CREATE PUBLIC DATABASE LINK DRDA2 CONNECT TO  
2 "user2" IDENTIFIED BY "password2" USING 'new_db2_using';
```

Enter the following to create a database link for the dg4drda3 gateway:

```
SQL> CREATE PUBLIC DATABASE LINK DRDA3 CONNECT TO  
2 "user3" IDENTIFIED BY "password3" USING 'new_db3_using';
```

After the database links are created you can verify the connection to the new DRDA databases, as in the following:

```
SQL> SELECT * FROM ALL_USERS@DRDA2;
```

```
SQL> SELECT * FROM ALL_USERS@DRDA3;
```

Security Considerations

The gateway architecture involves multiple computer setups that have distinct security capabilities and limitations. This chapter provides information for planning and implementing your security system.

It contains the following sections:

- [Security Overview](#)
- [Authenticating Application Logons](#)
- [Defining and Controlling Database Links](#)
- [Processing Inbound Connections](#)
- [Passwords in the Gateway Initialization File](#)

Security Overview

When you connect several different systems, generally the system with the strictest security requirements dictates and rules the system.

Gateway security involves two groups:

- Users and applications that are permitted access to a given gateway instance and DRDA database server
- Server database objects that users and applications are able to query and update

You can control access in the gateway architecture at several points. Control over database object access is provided by each DRDA database server with GRANTS and related native authorization mechanisms based on user ID.

When the gateway is involved in a SQL request, security mechanisms are in effect for each DRDA system component encountered by the gateway. The first system component encountered is the application tool or 3GL program. The last system component encountered is the DRDA database.

Authenticating Application Logons

An application must connect to an Oracle database before using the gateway. The type of logon authentication that you use determines the resulting Oracle user ID and can affect gateway operation. There are two basic types of authentication:

- Oracle authentication: With Oracle authentication, each Oracle user ID has a password known to Oracle database. When an application connects to the server, it supplies a user ID and password. Oracle database confirms that the user ID exists and that the password matches the one kept in the database.

- Operating system authentication: With operating system authentication, the servers underlying operating system is responsible for authentication. An Oracle user ID that is created with the `IDENTIFIED EXTERNALLY` attribute, instead of a password, is accessed with operating system authentication. To log into such a user ID, the application supplies a forward slash (/) for a user ID and does not supply a password.

To perform operating system authentication, the server determines the requester's operating system user ID, optionally adds a fixed prefix to it, and uses the result as the Oracle user ID. The server confirms that the user ID exists and is `IDENTIFIED EXTERNALLY`, but no password checking is done. The underlying assumption is that users were authenticated when they logged into the operating system.

Operating system authentication is not available on all platforms and is not available in some Oracle Net (client-server) and multi-threaded server configurations. Refer to the *Oracle Database Installation Guide 10g for UNIX Systems* and Oracle Net documentation to determine the availability of this feature.

For more information about authenticating application logons, refer to the *Oracle Database Reference*.

Defining and Controlling Database Links

The information here is specific to the gateway. For additional information on database links, refer to the *Oracle Database Reference*.

Link Accessibility

The database link should be accessible to a given user. A public database link can be used by any user ID. A private database link can be used only by the user who created it. The server makes no distinction regarding the type of use (such as read-only versus update or write) or accessibility of remote objects. The DRDA database, which is accessed, is responsible for these distinctions.

Links and CONNECT Clauses

The `CONNECT` clause is another security-related attribute of a database link. You can use the `CONNECT` clause to specify an explicit user ID and password, which can differ from the user's Oracle database user ID and password. This `CONNECT` user ID and password combination is sent to the gateway when the database link connection is first opened. Depending on gateway options, the gateway might send that user ID and password to the DRDA Server for validation.

If a database link is created without a `CONNECT` clause, then the user's Oracle database user ID and password are sent to the gateway when the connection is opened. If the user logs into the Oracle database with operating system authentication, then the gateway does not receive any user ID or password from the Oracle database. In this case, user ID mapping facilities at the DRDA Server can be used to make such a connection possible if all users on the same host can use the same DRDA database user ID.

Processing Inbound Connections

Current DRDA Servers provide options for manipulating the security conduct of an inbound (client) DRDA session request.

User ID Mapping

The most useful DRDA Server security capability is user ID mapping. User ID mapping refers to changing the user ID associated with an incoming DRDA request to some other user ID known to that server. This is a useful feature if your Oracle Database Gateway installation does not have a uniform user ID structure across all systems and databases.

DB2/OS390

The DB2 DDF Communication Database (CDB) stores inbound DRDA session security options.

These tables, pertinent to inbound sessions, have a role in security processing:

- `SYSIBM.IPNAMES` table

The `SYSIBM.IPNAMES` table controls inbound security conducted for TCP/IP based sessions, affecting all DRDA connections from a particular host system. This table also controls whether inbound connection user IDs are subject to translation or mapping.

- `SYSIBM.SYSUSERNAMES` table

When translation is used, rows in the `SYSIBM.SYSUSERNAMES` table specify translated user IDs by IP name and inbound user ID. Default entries that pertain to all IPs and to all inbound user IDs can be made in both tables. The mapping table can also be used simply to indicate which inbound user IDs are permitted from a particular IP or from all IPs, whether or not they are mapped.

This implementation provides a flexible mapping structure. You can specify that all connections from a particular IP use a single DB2 user ID, or that a particular inbound user ID always be mapped to a particular DB2 user ID regardless of origin. A `SYSUSERNAMES` entry with blank IP name and inbound user ID can designate a single default DB2 user ID for all connections unless a more specific entry, by IP name, user ID, or both, exists.

The CDB tables can be updated by a user with update privilege using a SQL tool such as the DB2 `SPUFI` utility. For example, most database administrators, systems programmers, and security officers can update CDB tables. The DB2 DDF component must be stopped and restarted for CDB changes to take effect.

The DB2 non-DRDA-specific security features are also involved in DRDA connections. User IDs are subject to normal DB2 or SAF/RACF validation in addition to connection or sign-on exit processing. Passwords are also subject to validation. After the connection is established, all normal authorizations or `GRANTS` associated with the user ID are in effect. The user ID must have execute privilege on the gateway DRDA package to process any SQL statements.

DB2/400

DB2/400 does not provide a user ID mapping capability comparable to that in DB2/OS390. Normally, the user ID in an incoming DRDA connection request must be a valid user ID on that DB2/400.

The DB2/400 subsystem communications entry for the gateway should specify that the gateway is not a secure location and should include a default user ID of `*NONE`.

After the application has completed the DRDA connection to the DB2/400, it is subject to all authorities and `GRANTS` associated with the user ID in use.

The user ID must have execute authority on the gateway DRDA package to execute any SQL statements.

DB2/Universal Database

DB2/Universal Database (DB2/UDB) does not provide a user ID mapping capability comparable to that in DB2/OS390. Normally, the user ID in an incoming DRDA connection request must be a valid user ID on the DB2/UDB host.

After the application has completed the DRDA connection to the DB2 host, it is subject to all authorities and GRANTS associated with the user ID in use. The user ID must have execute authority on the gateway DRDA package to execute any SQL statements.

Passwords in the Gateway Initialization File

The gateway uses user IDs and passwords to access the information in the remote database on the DRDA Server. Some user IDs and passwords must be defined in the gateway initialization file to handle functions such as resource recovery. In the current security conscious environment, having plain-text passwords that are accessible in the Initialization File is deemed insecure. An encryption feature has been added as part of Heterogeneous Services' generic connectivity to help make this more secure. This feature is accessible by this gateway. With it Initialization parameters which contain sensitive values might be stored in an encrypted form. Refer to Chapter 4, "Encrypting Initialization Parameters" of the Oracle Database Heterogeneous Connectivity Administrator's Guide, for information about how to use the feature.

See Also: the parameters `DRDA_RECOVERY_USERID` and `DRDA_RECOVERY_PASSWORD` in [Appendix C](#) as examples, for more information.

Migration From Previous Releases

This chapter describes how to migrate to new instances of Oracle Database Gateway for DRDA from an existing installation. Perform the following steps to migrate to a new release of Oracle Database Gateway for DRDA from an existing release:

1. [Install the New Release](#)
2. [Copy the Gateway Initialization Parameter File](#)
3. [Update the Initialization Parameters](#)
4. [Bind Gateway Package](#)
5. [Install/Upgrade Data Dictionary Views](#)

Install the New Release

Install the new release of the gateway in a separate directory, as discussed [Chapter 12, "Installing Oracle Database Gateway for DRDA"](#).

Caution: Do not install the gateway over a previously existing gateway installation. This corrupts the existing installation.

Copy the Gateway Initialization Parameter File

Copy the `initsid.ora` from the old gateway instance to the new instance. The formats of the parameters in the `initsid.ora` Gateway Initialization File have changed. Refer to "[Oracle Database Gateway for DRDA Initialization Parameters](#)" on page C-6.

Update the Initialization Parameters

The next step in migrating to a new release of Oracle Database Gateway for DRDA consists of updating the initialization parameters.

Changed Parameters

The use of `DRDA_CONNECT_PARM` has changed in this version. Refer to [Appendix C, "Initialization Parameters"](#) for the syntax of the parameters.

Obsolete Parameters

The following parameters are obsolete for the 11g version. Remove them from your configuration files:

- MODE
- SERVER_PATH
- DRDA_OVERRIDE_FROM_CODEPAGE
- DRDA_OVERRIDE_TO_CODEPAGE
- ERROR_LOGGING
- ERROR_REPORTING
- ERRORTAG
- GATEWAY_SID
- GROUP_BY_OFF
- GTWDEBUG
- INCREMENT_CURSORS
- DRDA_CALLDESC_STMT
- DRDA_CALLDESC_PROC

Bind Gateway Package

When upgrading to 11g release you must rebind the gateway package if you have changed any of the following initialization parameters:

- DRDA_DISABLE_CALL
- DRDA_ISOLATION_LEVEL
- DRDA_PACKAGE_COLLID
- DRDA_PACKAGE_CONSTOKEN
- DRDA_PACKAGE_NAME
- DRDA_PACKAGE_OWNER
- DRDA_PACKAGE_SECTIONS

Install/Upgrade Data Dictionary Views

You must install or upgrade the data dictionary views when upgrading the Oracle Database Gateway for DRDA. Refer to [Chapter 14, "Configuring Oracle Database Gateway for DRDA"](#) for more information on creating data dictionary views.

Part VIII

Removing Oracle Database Gateway

Part VIII, "Removing Oracle Database Gateway" describes how to remove Oracle Database Gateway.

It contains the following chapter:

- [Chapter 17, "Removing Oracle Database Gateway"](#)

Removing Oracle Database Gateway

If you decide to remove the Oracle Database Gateway, perform the following steps:

1. Stop the following process: Oracle Net Listener using the following command:

```
ORACLE_HOME/bin/lsnrctl stop
```

2. Log in as the oracle user:

```
$ su - oracle
```

3. Set the ORACLE_HOME environment variable to specify the path of the Oracle home that you want to remove:

- Bourne, Bash, or Korn shell:

```
$ ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1; export ORACLE_HOME
```

- C shell:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/11.1.0/db_1
```

4. Start the Installer as follows:

```
$ $ORACLE_HOME/oui/bin/runInstaller
```

5. Step through the Oracle Universal Installer. Use the prompts listed in [Table 17–1, "Steps to Deinstall the Oracle Database Gateway Using Oracle Universal Installer"](#) as a guide for removing, following the instructions in the Response column.

Table 17–1 Steps to Deinstall the Oracle Database Gateway Using Oracle Universal Installer

Prompt	Response
1. Oracle Universal Installer: Welcome	Click Deinstall Products .
2. Inventory	<p>You may either choose to remove <i>all</i> products that you may have installed at the time of your original gateway installation, or you may choose to remove only an instance of Oracle Database Gateway.</p> <ul style="list-style-type: none"> ■ To remove <i>all</i> products: Select the <i>ORACLE_HOME</i> where Oracle Database Gateway was installed. ■ Click Remove... ■ To remove <i>only</i> one instance of Oracle Database Gateway, expand the rows within the <i>ORACLE_HOME</i> recursively until you arrive at the folder. ■ Click Remove...

Table 17–1 (Cont.) Steps to Deinstall the Oracle Database Gateway Using Oracle Universal Installer

Prompt	Response
3. Confirmation	Click Yes .
4. Inventory	Click Close .
5. Oracle Universal Installer: Welcome	Click Cancel .
6. Exit	Click Yes .

6. The Oracle Database Gateway is now removed.

When the Oracle Universal Installer confirms that the deinstallation has ended, verify that the removal procedure was successful. To do this, read the contents of the deinstallation log file, which is located in the `C:\Program Files\Oracle\Inventory\logs` directory.

The default file name is `InstallActionsYYYY-MM-DD_HH-mm-SS-AM/PM.log`, where:

YYYY is year;

MM is month

DD is day

HH is hour

mm is minute

SS is seconds

AM/PM is daytime or evening

These variables in the log file name represent the date and time the product was removed.

7. The only files that are removed are those that were copied to the `ORACLE_HOME` directory during the gateway installation. You must remove any other related files manually, including deleting `listener.ora` and `tnsnames.ora` entries relating to the gateway, dropping database links.

Part IX

Appendixes

Part IX, "Appendixes" includes appendixes containing information relevant to installing and configuring Oracle Database Gateways.

It contains the following chapters:

- [Appendix A, "Using Response Files for Noninteractive Installation"](#)
- [Appendix B, "Oracle Database Gateway Troubleshooting"](#)
- [Appendix C, "Initialization Parameters"](#)
- [Appendix D, "Configuration Worksheet for DRDA"](#)
- [Appendix E, "Globalization Support for DRDA"](#)

Using Response Files for Noninteractive Installation

This appendix describes how to install and configure Oracle products using response files. It includes information about the following topics:

- [Introduction](#)
- [Creating the oraInst.loc File](#)
- [Preparing a Response File](#)
- [Running Oracle Universal Installer in Silent or Suppressed Mode](#)

Introduction

You can automate the installation and configuration of Oracle software, either fully or partially, by specifying a response file when you start Oracle Universal Installer. Oracle Universal Installer uses the values contained in the response file to provide answers to some or all of Oracle Universal Installer prompts:

- If you include responses for all of the prompts in the response file and specify the `-silent` option when starting Oracle Universal Installer, then Oracle Universal Installer runs in silent mode. During a silent-mode installation, Oracle Universal Installer does not display any screens. Instead, it displays progress information in the terminal that you used to start it.
- If you include responses for some or all of the prompts in the response file and omit the `-silent` option, then Oracle Universal Installer runs in suppressed mode. During a suppressed-mode installation, Oracle Universal Installer displays only the screens for which you did not specify all required information. You can also use variables in the response file or command-line options to suppress other installer screens, such as the Welcome screen or Summary screen, that do not prompt for information.

The following table describes several reasons why you might want to run Oracle Universal Installer in silent mode or suppressed mode:

Mode	Uses
Silent	<p>Use silent mode if you want to:</p> <ul style="list-style-type: none"> ■ Complete an unattended installation, which you might schedule using operating system utilities such as <code>at</code> ■ Complete several similar installations on multiple systems without user interaction ■ Install the software on a system that does not have X Window System software installed on it <p>Oracle Universal Installer displays progress information in the terminal that you used to start it, but it does not display any of Oracle Universal Installer screens.</p>
Suppressed	<p>Use suppressed mode if you want to complete similar Oracle software installations on more than one system, providing default answers to some, but not all of Oracle Universal Installer prompts.</p> <p>If you do not specify information required for a particular Installer screen in the response file, Oracle Universal Installer displays that screen. It suppresses screens for which you have provided all of the required information.</p>

Installation Overview

To install and configure Oracle products using Oracle Universal Installer in silent or suppressed mode, follow these steps:

1. Create the `oraInst.loc` file.
2. Prepare a response file.
3. Run Oracle Universal Installer in silent or suppressed mode.

These steps are described in the following sections.

Creating the oraInst.loc File

If you plan to install Oracle products using Oracle Universal Installer in silent or suppressed mode, you must manually create the `oraInst.loc` file if it does not already exist. This file specifies the location of the Oracle Inventory directory where Oracle Universal Installer creates the inventory of Oracle products installed on the system.

Note: If Oracle software has been installed previously on the system, the `oraInst.loc` file might already exist. If the file does exist, you do not need to create a file.

To create the `oraInst.loc` file, follow these steps:

1. Switch user to root:

2. On HP-UX PA-RISC and Solaris (SPARC), create the `/var/opt/oracle` directory if it does not exist:

```
# mkdir /var/opt/oracle
```

3. Change directory as follows, depending on your operating system:

AIX:

```
# cd /etc
```

HP-UX PA-RISC and Solaris (SPARC):

```
# cd /var/opt/oracle
```

4. Enter the following commands to set the appropriate owner, group, and permissions on the `oraInst.loc` file:

```
# chown oracle:oinstall oraInst.loc
# chmod 664 oraInst.loc
```

Preparing a Response File

This section describes the methods that you can use to prepare a response file for use during silent-mode or suppressed-mode installations:

- [Editing a Response File Template](#)
- [Recording a Response File](#)

Editing a Response File Template

Oracle provides response file templates for each product and installation type, and for each configuration tool. The response files for Oracle Gateways, `tg.rsp` and `netca.rsp` are located in the `response` directory on the media.

Note: If you copied the software to a hard disk, the response files are located in the `Disk1/response` directory.

To prepare a response file:

1. Copy the response file from the response file directory to a directory on your system:

```
$ cp /directory_path/response/response_file.rsp local_directory
```

In this example, `directory_path` is the CD-ROM mount point directory or the directory on the DVD. If you have copied the software to a hard drive, you can edit the file in the `response` directory if you prefer.

2. Open the response file in a text editor:

```
$ vi /local_dir/response_file.rsp
```

3. Edit the file, following the instructions in the file.

Note: Oracle Universal Installer or configuration assistant fails if you do not correctly configure the response file. Refer to the "[Silent-Mode Response File Error Handling](#)" section on page B-3 for more information about troubleshooting a failed silent-mode installation.

4. Change the permissions on the file to 700:

```
$ chmod 700 /local_dir/response_file.rsp
```

Recording a Response File

This method is most useful for Custom or software-only installations.

You can use Oracle Universal Installer in interactive mode to record a response file that you can edit and then use to complete silent-mode or suppressed-mode installations. When you are recording the response file, you can either complete the installation, or you can exit from Oracle Universal Installer on the Summary page, before it starts to copy the software to the system.

To record a new response file:

1. Complete the pre-installation tasks listed in respective chapters.

When you run Oracle Universal Installer to record a response file, it checks the system to verify that it meets the requirements to install the software. For this reason, Oracle recommends that you complete all of the required pre-installation tasks and record the response file while completing an installation.

2. If you have not installed Oracle software on this system previously, create the `oraInst.loc` file, as described in the previous section.
3. Ensure that the Oracle software owner user (typically `oracle`) has permissions to create or write to the Oracle home path that you will specify when you run Oracle Universal Installer.
4. To record a response file, enter a command similar to the following to start Oracle Universal Installer:

Note: Do not specify a relative path to the response file. If you specify a relative path, Oracle Universal Installer fails.

```
$ /directory_path/runInstaller -record -destinationFile filename
```

In the previous example:

- *directory_path* is either the CD-ROM mount point directory, the path of the directory on the DVD, or the path of the `disk1` directory on the hard drive
 - The `-record` parameter specifies that you want to record the responses that you enter in a response file
 - *filename* is the full path and file name of the response file that you want to record
5. On each Installer screen, specify the required information.
 6. When Oracle Universal Installer displays the Summary screen, do one of the following:
 - Click **Install** to create the response file, then continue with the installation.
 - Click **Cancel**, then **Yes** to create the response file but exit from Oracle Universal Installer without installing the software.

The response file is saved in the location that you specified using the `-destinationFile` option.

7. If you did not complete the installation, delete the Oracle home directory that Oracle Universal Installer created using the path you specified on the Specify File Locations screen.

8. Before using the recorded response file on another system, use a text editor to edit the file and make any required changes.

Use the comments in the file as a guide when editing it.

Running Oracle Universal Installer in Silent or Suppressed Mode

To run Oracle Universal Installer in silent or suppressed mode, follow these steps:

1. Complete the pre-installation tasks listed in the respective chapters.
2. Log in as the Oracle software owner user (typically `oracle`).
3. To start Oracle Universal Installer in silent or suppressed mode, enter a command similar to the following:

```
$ $ /directory_path/runInstaller -silent -noconfig -responseFile filename
```

Note: Do not specify a relative path to the response file. If you specify a relative path, Oracle Universal Installer fails.

In this example:

- *directory_path* is either the installation media mount point directory, the path of the directory on the DVD, or the path of the `Disk1` directory on the hard drive.
- `-silent` indicates that you want to run Oracle Universal Installer in silent mode.
- `-noconfig` suppresses running the configuration assistants during installation, and a software-only installation is performed instead.
- *filename* is the full path and file name of the installation response file that you configured.

Note: For more information about other options for the `runInstaller` command, enter the following command:

```
$ /directory_path/runInstaller -help
```

Oracle Database Gateway Troubleshooting

This appendix contains information about troubleshooting. It includes information about the following topics:

- [Verify Requirements](#)
- [What to Do If an Installation Error Occurs](#)
- [Reviewing the Log of an Installation Session](#)
- [Troubleshooting Configuration Assistants](#)
- [Silent-Mode Response File Error Handling](#)
- [Cleaning Up After a Failed Installation](#)

Verify Requirements

Before performing any of the troubleshooting steps in this appendix, ensure that the system meets the requirements and that you have completed all of the pre-installation tasks specified in respective chapters.

Read the Release Notes

Read the release notes for the product before installing it. The release notes are available on the Oracle Database 11g installation media. The latest version of the release notes is also available on the OTN Web site:

<http://www.oracle.com/technology/documentation/index.html>

What to Do If an Installation Error Occurs

If you encounter an error during installation:

- Do not exit Oracle Universal Installer.
- If you clicked **Next** after you entered incorrect information on one of the installation screens, click **Back** to return to the screen and correct the information.
- If you encounter an error while Oracle Universal Installer is copying or linking files, refer to the "[Reviewing the Log of an Installation Session](#)" section on page B-2.
- If you encounter an error while a configuration assistant is running, refer to the "[Troubleshooting Configuration Assistants](#)" section on page B-2.
- If you cannot resolve the problem, remove the failed installation by following the steps listed in the "[Cleaning Up After a Failed Installation](#)" section on page B-4.

Reviewing the Log of an Installation Session

During an installation, Oracle Universal Installer records all of the actions that it performs in a log file. If you encounter problems during the installation, review the log file for information about possible causes of the problem.

To view the log file, follow these steps:

1. If necessary, enter the following command to determine the location of the `oraInventory` directory:

For AIX and Linux:

```
$ cat /etc/oraInst.loc
```

For Solaris SPARC:

```
# more /var/opt/oracle/oraInst.loc
```

For HP-UX PA-RISC:

```
$ cat /var/opt/oracle/oraInst.loc
```

The `inventory_loc` parameter in this file specifies the location of the `oraInventory` directory.

2. Enter the following command to change directory to Oracle Universal Installer log file directory, where `orainventory_location` is the location of the `oraInventory` directory:

```
$ cd /orainventory_location/logs
```

3. Enter the following command to determine the name of the log file:

```
$ ls -ltr
```

This command lists the files in the order of creation, with the most recent file shown last. Installer log files have names similar to the following, where `date_time` indicates the date and time that the installation started:

```
installActionsdate_time.log
```

4. To view the most recent entries in the log file, where information about a problem is most likely to appear, enter a command similar to the following:

```
$ tail -50 installActionsdate_time.log | more
```

This command displays the last 50 lines in the log file.

5. If the error displayed by Oracle Universal Installer or listed in the log file indicates a relinking problem, refer to the following file for more information:

```
$(ORACLE_HOME)/install/make.log
```

Troubleshooting Configuration Assistants

To troubleshoot an installation error that occurs when a configuration assistant is running:

- Review the installation log files listed in the ["Reviewing the Log of an Installation Session"](#) section on page B-2.

- Review the specific configuration assistant log file located in the `$ORACLE_HOME/cfgtoollogs` directory. Try to fix the issue that caused the error.
- If you see the "Fatal Error. Reinstall" message, look for the cause of the problem by reviewing the log files. Refer to the ["Fatal Errors"](#) section on page B-3 for further instructions.

Configuration Assistant Failure

Oracle configuration assistant failures are noted at the bottom of the installation screen. The configuration assistant interface displays additional information, if available. The configuration assistant execution status is stored in the following file:

`oraInventory_location/logs/installActionsdate_time.log`

The execution status codes are listed in the following table:

Status	Result Code
Configuration assistant succeeded	0
Configuration assistant failed	1
Configuration assistant cancelled	-1

Fatal Errors

If you receive a fatal error while a configuration assistant is running, you must remove the current installation and reinstall the Oracle software, as follows:

1. Remove the failed installation as described in the ["Cleaning Up After a Failed Installation"](#) section on page B-4.
2. Correct the cause of the fatal error.
3. Reinstall the Oracle software.

Silent-Mode Response File Error Handling

To determine whether a silent-mode installation succeeds or fails, refer to the following log file:

`/oraInventory_location/logs/silentInstalldate_time.log`

If necessary, refer to the previous section for information about determining the location of the `oraInventory` directory.

A silent installation fails if:

- You do not specify a response file
- You specify an incorrect or incomplete response file
- Oracle Universal Installer encounters an error, such as insufficient disk space

Oracle Universal Installer or configuration assistant validates the response file at run time. If the validation fails, the silent-mode installation or configuration process ends. Oracle Universal Installer treats values for parameters that are of the wrong context, format, or type as if no value was specified in the file.

Cleaning Up After a Failed Installation

If an installation fails, you must remove files that Oracle Universal Installer created during the attempted installation and remove the Oracle home directory. Perform the following steps to remove the files:

1. Start Oracle Universal Installer as described in the "[Running the Oracle Universal Installer](#)" section on page 1-5.
2. Click **Deinstall Products** on the Welcome window or click **Installed Products** on any Installer window.

The Inventory window appears, listing installed products.

3. Select the Oracle home that contains the products that you want to remove, then click **Remove**.
4. Manually remove the Oracle home directory created during the failed installation.
5. Reinstall the Oracle software.

Initialization Parameters

The Oracle database initialization parameters in the `init.ora` file are distinct from gateway initialization parameters. Set the gateway parameters in the initialization parameter file using an agent-specific mechanism, or set them in the Oracle data dictionary using the `DBMS_HS` package. The gateway initialization parameter file must be available when the gateway is started. Changes made to the initialization parameters only take effect in the next gateway session.

This appendix contains a list of the gateway initialization parameters that can be set for each gateway and their description. It also describes the initialization parameter file syntax. It includes the following sections:

- [Initialization Parameter File Syntax](#)
- [Oracle Database Gateway for Sybase Initialization Parameters](#)
- [Oracle Database Gateway for Informix Initialization Parameters](#)
- [Oracle Database Gateway for Teradata Initialization Parameters](#)
- [Oracle Database Gateway for SQL Server Initialization Parameters](#)
- [Oracle Database Gateway for ODBC Initialization Parameters](#)
- [Oracle Database Gateway for DRDA Initialization Parameters](#)
- [Initialization Parameter Descriptions](#)

Initialization Parameter File Syntax

The syntax for the initialization parameter file is as follows:

1. The file is a sequence of commands.
2. Each command should start on a separate line.
3. End of line is considered a command terminator (unless escaped with a backslash).
4. If there is a syntax error in an initialization parameter file, none of the settings take effect.
5. Set the parameter values as follows:

```
[SET] [PRIVATE] parameter=value
```

Where:

parameter is an initialization parameter name. It is a string of characters starting with a letter and consisting of letters, digits and underscores. Initialization parameter names are case sensitive.

value is the initialization parameter value. It is case-sensitive. An initialization parameter value is either:

- a. A string of characters that does not contain any backslashes, white space or double quotation marks (")
- b. A quoted string beginning with a double quotation mark and ending with a double quotation mark. The following can be used inside a quoted string:
 - * backslash (\) is the escape character
 - * \n inserts a new line
 - * \t inserts a tab
 - * \" inserts a double quotation mark
 - * \\ inserts a backslash

A backslash at the end of the line continues the string on the next line. If a backslash precedes any other character then the backslash is ignored.

For example, to enable tracing for an agent, set the HS_FDS_TRACE_LEVEL initialization parameter as follows:

```
HS_FDS_TRACE_LEVEL=ON
```

SET and PRIVATE are optional keywords. You cannot use either as an initialization parameter name. Most parameters are needed only as initialization parameters, so you usually do not need to use the SET or PRIVATE keywords. If you do not specify either SET or PRIVATE, the parameter is used only as an initialization parameter for the agent.

SET specifies that, in addition to being used as an initialization parameter, the parameter value is set as an environment variable for the agent process. Use SET for parameter values that the drivers or non-Oracle system need as environment variables.

PRIVATE specifies that the initialization parameter should be private to the agent and should not be uploaded to the Oracle database. Most initialization parameters should not be private. If, however, you are storing sensitive information like a password in the initialization parameter file, then you may not want it uploaded to the server because the initialization parameters and values are not encrypted when uploaded. Making the initialization parameters private prevents the upload from happening and they do not appear in dynamic performance views. Use PRIVATE for the initialization parameters only if the parameter value includes sensitive information such as a username or password.

SET PRIVATE specifies that the parameter value is set as an environment variable for the agent process and is also private (not transferred to the Oracle database, not appearing in dynamic performance views or graphical user interfaces).

Oracle Database Gateway for Sybase Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for Sybase. They are as follows:

- [HS_CALL_NAME](#)

- HS_DB_DOMAIN
- HS_DB_INTERNAL_NAME
- HS_DB_NAME
- HS_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TIME_ZONE
- HS_TRANSACTION_MODEL
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER
- HS_FDS_PROC_IS_FUNC
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_RESULTSET_SUPPORT
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_FETCH_ROWS

Oracle Database Gateway for Informix Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for Informix. They are as follows:

- HS_DB_DOMAIN
- HS_DB_INTERNAL_NAME
- HS_DB_NAME
- HS_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TIME_ZONE
- HS_TRANSACTION_MODEL
- IFILE

- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_FETCH_ROWS

Oracle Database Gateway for Teradata Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for Teradata. They are as follows:

- HS_DB_DOMAIN
- HS_DB_INTERNAL_NAME
- HS_DB_NAME
- HS_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TIME_ZONE
- HS_TRANSACTION_MODEL
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_FETCH_ROWS

Oracle Database Gateway for SQL Server Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for Microsoft SQL Server. They are as follows:

- HS_CALL_NAME
- HS_DB_DOMAIN
- HS_DB_INTERNAL_NAME
- HS_DB_NAME

- HS_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TIME_ZONE
- HS_TRANSACTION_MODEL
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER
- HS_FDS_PROC_IS_FUNC
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_RESULTSET_SUPPORT
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_REPORT_REAL_AS_DOUBLE
- HS_FDS_FETCH_ROWS

Oracle Database Gateway for ODBC Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for ODBC. They are as follows:

- HS_DB_DOMAIN
- HS_DB_INTERNAL_NAME
- HS_DB_NAME
- HS_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_FDS_SHAREABLE_NAME
- HS_TIME_ZONE
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER

- HS_FDS_TRACE_LEVEL
- HS_TRANSACTION_MODEL
- HS_FDS_FETCH_ROWS

Oracle Database Gateway for DRDA Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for DRDA. They are as follows:

- HS_CALL_NAME
- HS_DB_DOMAIN
- HS_DB_INTERNAL_NAME
- HS_DB_NAME
- HS_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TRANSACTION_MODEL
- HS_FDS_FETCH_ROWS
- IFILE
- DRDA_CACHE_TABLE_DESC
- DRDA_CAPABILITY
- DRDA_CODEPAGE_MAP
- DRDA_COMM_BUFLLEN
- DRDA_CONNECT_PARM
- DRDA_DEFAULT_CCSID
- DRDA_DESCRIBE_TABLE
- DRDA_DISABLE_CALL
- DRDA_FLUSH_CACHE
- DRDA_GRAPHIC_CHAR_SIZE
- DRDA_GRAPHIC_PAD_SIZE
- DRDA_GRAPHIC_LIT_CHECK
- DRDA_GRAPHIC_TO_MBCS
- DRDA_ISOLATION_LEVEL
- DRDA_LOCAL_NODE_NAME
- DRDA_MBCS_TO_GRAPHIC
- DRDA_OPTIMIZE_QUERY
- DRDA_PACKAGE_COLLID

- DRDA_PACKAGE_CONSTOKEN
- DRDA_PACKAGE_NAME
- DRDA_PACKAGE_OWNER
- DRDA_PACKAGE_SECTIONS
- DRDA_READ_ONLY
- DRDA_RECOVERY_PASSWORD
- DRDA_RECOVERY_USERID
- DRDA_REMOTE_DB_NAME
- FDS_CLASS
- HS-NLS_NCHAR
- LOG_DESTINATION
- ORA_MAX_DATE
- ORA-NLS11
- ORACLE_DRDA_TCTL
- ORACLE_DRDA_TRACE
- TRACE_LEVEL
- HS-NLS_DATE_FORMAT
- HS-NLS_DATE_LANGUAGE
- HS-NLS_NUMERIC_CHARACTER

Initialization Parameter Description

The following sections describe all the initialization file parameters that can be set for gateways.

HS_CALL_NAME

Property	Description
Default value	None
Range of values	Not applicable

Specifies the remote functions that can be referenced in SQL statements. The value is a list of remote functions and their owners, separated by semicolons, in the following format:

owner_name.function_name

For example:

`owner1.A1;owner2.A2;owner3.A3`

If an owner name is not specified for a remote function, the default owner name becomes the user name used to connect to the remote database (specified when the Heterogeneous Services database link is created or taken from user session if not specified in the DB link).

The entries for the owner names and the function names are case-sensitive.

HS_DB_DOMAIN

Property	Description
Default value	WORLD
Range of values	1 to 199 characters

Specifies a unique network sub-address for a non-Oracle system. The HS_DB_DOMAIN initialization parameter is similar to the DB_DOMAIN initialization parameter, described in the *Oracle Database Reference*. The HS_DB_DOMAIN initialization parameter is required if you use the Oracle Names server. The HS_DB_NAME and HS_DB_DOMAIN initialization parameters define the global name of the non-Oracle system.

Note: The HS_DB_NAME and HS_DB_DOMAIN initialization parameters must combine to form a unique address in a cooperative server environment.

HS_DB_INTERNAL_NAME

Property	Description
Default value	01010101
Range of values	1 to 16 hexadecimal characters

Specifies a unique hexadecimal number identifying the instance to which the Heterogeneous Services agent is connected. This parameter's value is used as part of a transaction ID when global name services are activated. Specifying a nonunique number can cause problems when two-phase commit recovery actions are necessary for a transaction.

HS_DB_NAME

Property	Description
Default value	HO
Range of values	1 to 8 characters

Specifies a unique alphanumeric name for the data store given to the non-Oracle system. This name identifies the non-Oracle system within the cooperative server environment. The HS_DB_NAME and HS_DB_DOMAIN initialization parameters define the global name of the non-Oracle system.

HS_DESCRIBE_CACHE_HWM

Property	Description
Default value	100

Property	Description
Range of values	1 to 4000

Specifies the maximum number of entries in the describe cache used by Heterogeneous Services. This limit is known as the describe cache high water mark. The cache contains descriptions of the mapped tables that Heterogeneous Services reuses so that it does not have to re-access the non-Oracle data store.

If you are accessing many mapped tables, increase the high water mark to improve performance. Increasing the high water mark improves performance at the cost of memory usage.

HS_LANGUAGE

Property	Description
Default value	System-specific
Range of values	Any valid language name (up to 255 characters)

Provides Heterogeneous Services with character set, language, and territory information of the non-Oracle data source. The value must use the following format:

language[_territory.character_set]

Note: The globalization support initialization parameters affect error messages, the data for the SQL Service, and parameters in distributed external procedures.

Character Sets

Ideally, the character sets of the Oracle database and the non-Oracle data source are the same. If they are not the same, Heterogeneous Services attempts to translate the character set of the non-Oracle data source to the Oracle database character set, and back again. The translation can degrade performance. In some cases, Heterogeneous Services cannot translate a character from one character set to another.

Note: The specified character set must be a superset of the operating system character set on the platform where the agent is installed.

Language

The language component of the HS_LANGUAGE initialization parameter determines:

- Day and month names of dates
- AD, BC, PM, and AM symbols for date and time
- Default sorting mechanism

Note that Oracle does not determine the language for error messages for the generic Heterogeneous Services messages (ORA-25000 through ORA-28000). These are controlled by the session settings in the Oracle database.

Note: Use the `HS_NLS_DATE_LANGUAGE` initialization parameter to set the day and month names, and the AD, BC, PM, and AM symbols for dates and time independently from the language.

Territory

The territory clause specifies the conventions for day and week numbering, default date format, decimal character and group separator, and ISO and local currency symbols. Note that the level of globalization support between the Oracle database and the non-Oracle data source depends on how the gateway is implemented.

HS_LONG_PIECE_TRANSFER_SIZE

Property	Description
Default value	64 KB
Range of values	Any value up to 2 GB

Sets the size of the piece of `LONG` data being transferred. A smaller piece size means less memory requirement, but more round-trips to fetch all the data. A larger piece size means fewer round-trips, but more of a memory requirement to store the intermediate pieces internally. Thus, the initialization parameter can be used to tune a system for the best performance, with the best trade-off between round-trips and memory requirements, and network latency or response time.

HS_OPEN_CURSORS

Property	Description
Default value	50
Range of values	1 to the value of <code>OPEN_CURSORS</code> initialization parameter of Oracle database

Defines the maximum number of cursors that can be open on one connection to a non-Oracle system instance.

The value never exceeds the number of open cursors in the Oracle database. Therefore, setting the same value as the `OPEN_CURSORS` initialization parameter in the Oracle database is recommended.

HS_RPC_FETCH_REBLOCKING

Property	Description
Default value	ON
Range of values	OFF or ON

Controls whether Heterogeneous Services attempts to optimize performance of data transfer between the Oracle database and the Heterogeneous Services agent connected to the non-Oracle data store.

The following values are possible:

- OFF disables reblocking of fetched data so that data is immediately sent from agent to server.
- ON enables reblocking, which means that data fetched from the non-Oracle system is buffered in the agent and is not sent to the Oracle database until the amount of fetched data is equal or higher than the value of `HS_RPC_FETCH_SIZE` initialization parameter. However, any buffered data is returned immediately when a fetch indicates that no more data exists or when the non-Oracle system reports an error.

HS_RPC_FETCH_SIZE

Property	Description
Default value	50000
Range of values	1 to 10000000

Tunes internal data buffering to optimize the data transfer rate between the server and the agent process.

Increasing the value can reduce the number of network round-trips needed to transfer a given amount of data, but also tends to increase data bandwidth and to reduce latency as measured between issuing a query and completion of all fetches for the query. Nevertheless, increasing the fetch size can increase latency for the initial fetch results of a query, because the first fetch results are not transmitted until additional data is available.

HS_TIME_ZONE

Property	Description
Default value for '[+ -]hh:mm'	Derived from the <code>NLS_TERRITORY</code> initialization parameter
Range of values for '[+ -]hh:mm'	Any valid datetime format mask

Specifies the default local time zone displacement for the current SQL session. The format mask, [+|-]hh:mm, is specified to indicate the hours and minutes before or after UTC (Coordinated Universal Time—formerly Greenwich Mean Time). For example:

```
HS_TIME_ZONE = [+ | -] hh:mm
```

HS_TRANSACTION_MODEL

Property	Description
Default Value	<code>COMMIT_CONFIRM</code>
Range of Values	<code>COMMIT_CONFIRM</code> , <code>READ_ONLY</code> , <code>SINGLE_SITE</code>

Specifies the type of transaction model that is used when the non-Oracle database is updated by a transaction.

The following values are possible:

- COMMIT_CONFIRM provides read and write access to the non-Oracle database and allows the gateway to be part of a distributed update. To use the commit-confirm model, the following items must be created in the non-Oracle database:
 - Transaction log table. The default table name is HS_TRANSACTION_LOG. A different name can be set using the HS_FDS_TRANSACTION_LOG parameter. The transaction log table must be granted SELECT, DELETE, and INSERT privileges set to public.
 - Recovery account. The account name is assigned with the HS_FDS_RECOVERY_ACCOUNT parameter.
 - Recovery account password. The password is assigned with the HS_FDS_RECOVERY_PWD parameter.

COMMIT_CONFIRM does not apply to Oracle Database Gateway for ODBC. The default value for Oracle Database Gateway for ODBC is SINGLE_SITE.
- READ_ONLY provides read access to the non-Oracle database.
- SINGLE_SITE provides read and write access to the non-Oracle database. However, the gateway cannot participate in distributed updates.

IFILE

Property	Description
Default value	None
Range of values	Valid parameter file names

Use the IFILE initialization parameter to embed another initialization file within the current initialization file. The value should be an absolute path and should not contain environment variables. The three levels of nesting limit does not apply.

See Also: *Oracle Database Reference*

HS_FDS_CONNECT_INFO

Property	Description
Default Value	None
Range of Values	Not applicable

HS_FDS_CONNECT_INFO which describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. The syntax for HS_FDS_CONNECT_INFO for the gateways are as follows:

For Oracle Database Gateway for Sybase:

`HS_FDS_CONNECT_INFO=host_name:port_number/database_name`

where, *host_name* is the host name or IP address of the machine hosting the Sybase database, *port_number* is the port number of the Sybase database server, and *database_name* is the Sybase database name.

For Oracle Database Gateway for Informix:

```
HS_FDS_CONNECT_INFO=host_name:port_number/server_name/database_name
```

where, *host_name* is the host name or IP address of the machine hosting the Informix database, *port_number* is the port number of the Informix database server, *server_name* is the name of the server machine for the Informix data, and *database_name* is the Informix database name.

For Oracle Database Gateway for Teradata:

```
HS_FDS_CONNECT_INFO=host_alias:port_number[/database_name]
```

where, *host_alias* is the host alias name or IP address of the machine hosting the Teradata database, *port_number* is the port number of the Teradata database server, and *database_name* is the Teradata database name. The *database_name* variable is optional.

For Oracle Database Gateway for SQL Server:

```
HS_FDS_CONNECT_INFO=host_name[:port_number][/[instance_name]][/database_name]
```

where, *host_name* is the host name or IP address of the machine hosting the SQL Server database, *port_number* is the port number of the SQL Server database server, *instance_name* is the instance of SQL Server running on the machine, and *database_name* is the SQL Server database name. Either of the variables *port_number* or *instance_name* can be used, but not both together. Optionally, they both can be omitted. The variable *database_name* is always optional. The slash (/) is required when a particular value is omitted. For example, all of the following entries are valid:

```
HS_FDS_CONNECT_INFO=host_name/instance_name/database_name
HS_FDS_CONNECT_INFO=host_name//database_name
HS_FDS_CONNECT_INFO=host_name:port_name//database_name
HS_FDS_CONNECT_INFO=host_name/instance_name
HS_FDS_CONNECT_INFO=host_name
```

For Oracle Database Gateway for ODBC:

```
HS_FDS_CONNECT_INFO=dsn_value
```

where *dsn_value* is the data source name configured in the `odbc.ini` file.

HS_FDS_DEFAULT_OWNER

Property	Description
Default Value	None
Range of Values	Not applicable

The name of the table owner that is used for the non-Oracle database tables if an owner is not specified in the SQL statements.

Note: If this parameter is not specified and the owner is not explicitly specified in the SQL statement, then the user name of the Oracle user or the user name specified when creating the database link is used.

HS_FDS_PROC_IS_FUNC

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE

Enables return values from functions. By default, all stored procedures and functions do not return a return value to the user.

Note: If you set this initialization parameter, you must change the syntax of the procedure execute statement for all existing stored procedures to handle return values.

HS_FDS_RECOVERY_ACCOUNT

Property	Description
Default Value	RECOVER
Range of values	Any valid user ID

Specifies the name of the recovery account used for the commit-confirm transaction model. An account with user name and password must be set up at the non-Oracle system. For more information about the commit-confirm model, see the HS_TRANSACTION_MODEL parameter.

The name of the recovery account is case-sensitive.

HS_FDS_RECOVERY_PWD

Property	Description
Default Value	RECOVER
Range of values	Any valid password

Specifies the password of the recovery account used for the commit-confirm transaction model set up at the non-Oracle system. For more information about the commit-confirm model, see the HS_TRANSACTION_MODEL parameter.

The name of the password of the recovery account is case-sensitive.

HS_FDS_RESULTSET_SUPPORT

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE

Enables result sets to be returned from stored procedures. By default, all stored procedures do not return a result set to the user.

Note: If you set this initialization parameter, you must do the following:

- Change the syntax of the procedure execute statement for all existing stored procedures, to handle result sets
 - Work in the sequential mode of Heterogeneous Services
-

HS_FDS_TRACE_LEVEL

Property	Description
Default Value	OFF
Range of values	OFF, ON, DEBUG

Specifies whether error tracing is turned on or off for gateway connectivity.

The following values are valid:

- OFF disables the tracing of error messages.
- ON enables the tracing of error messages that occur when you encounter problems. The results are written by default to a gateway log file in LOG directory where the gateway is installed.
- DEBUG enables the tracing of detailed error messages that can be used for debugging.

HS_FDS_TRANSACTION_LOG

Property	Description
Default Value	HS_TRANSACTION_LOG
Range of Values	Any valid table name

Specifies the name of the table created in the non-Oracle system for logging transactions. For more information about the transaction model, see the HS_TRANSACTION_MODEL parameter.

HS_FDS_SHAREABLE_NAME

Property	Description
Default Value	None
Range of Values	Not applicable

Specifies the full path name to the ODBC driver manager.

This is a required parameter, whose format is:

```
HS_FDS_SHAREABLE_NAME=odbc_installation_path/lib/libodbc.sl
```

Where:

odbc_installation_path is the path where the ODBC driver is installed.

HS_FDS_REPORT_REAL_AS_DOUBLE

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE

Enables Oracle Database Gateway for SQL Server to treat SINGLE FLOAT PRECISION fields as DOUBLE FLOAT PRECISION fields.

HS_FDS_FETCH_ROWS

Property	Description
Default Value	100
Range of Values	Any integer between 1 and 1000
Syntax	HS_FDS_FETCH_ROWS= <i>num</i>

HS_FDS_FETCH_ROWS specifies the fetch array size. This is the number of rows to be fetched from the non-Oracle database and to return to Oracle database at one time. This parameter will be affected by the HS_RPC_FETCH_SIZE and HS_RPC_FETCH_REBLOCKING parameters.

DRDA_CACHE_TABLE_DESC

Property	Description
Default Value	TRUE
Range of Values	{TRUE FALSE}
Syntax	DRDA_CACHE_TABLE_DESC= { <i>TRUE</i> / <i>FALSE</i> }

DRDA_CACHE_TABLE_DESC directs the gateway to cache table descriptions once per transaction. This can reduce the number of table lookups requested by Oracle database and can speed up the execution of SQL statements. You may wish to disable this

option if you would be altering the structure of a remote table and if you would be examining it within the same transaction.

DRDA_CAPABILITY

Property	Description
Default Value	None
Range of Values	Refer to Chapter 4, "Developing Applications" in <i>Oracle Database Gateway for DRDA User's Guide</i>
Syntax	DRDA_CAPABILITY={ <i>FUNCTION</i> /{ <i>ON</i> / <i>OFF</i> }}, . . .

DRDA_CAPABILITY specifies which mapped functions of Oracle database will be treated natively. In other words, no special pre processing or post processing will be done for these functions. They will be passed through to the DRDA Server unmodified.

DRDA_CODEPAGE_MAP

Property	Description
Default Value	codepage.map
Range of Values	Any valid file path
Syntax	DRDA_CODEPAGE_MAP= <i>codepage.map</i>

DRDA_CODEPAGE_MAP specifies the location of the codepage map. You may specify only the filename, which will be searched for within the \$ORACLE_HOME/dg4drda/admin directory, or you may specify the full path name of the file.

DRDA_COMM_BUFLLEN

Property	Description
Default Value	32767
Range of Values	512 through 32767
Syntax	DRDA_COMM_BUFLLEN= <i>num</i>

DRDA_COMM_BUFLLEN specifies the communications buffer length. This is a number indicating the TCP/IP buffer size in bytes.

DRDA_CONNECT_PARM

Property	Description
Default Value	DRDACON1:446
Range of Values	Any alphanumeric string 1 to 255 characters in length
Syntax	DRDA_CONNECT_PARM={ <i>hostname</i> / <i>ip_address</i> }{: <i>port</i> }

DRDA_CONNECT_PARM specifies the TCP/IP hostname or IP Address of the DRDA Server and, as an option, the Service Port number on which the DRDA Server is listening.

The DRDA standard specifies that port 446 be used for DRDA services. However, if several DRDA servers are operating on the same system, then they will need to provide service on different ports. Therefore, the port number that is used by each DRDA server will need to be extracted from the configuration of each individual DRDA server. DB2 for OS/390 and DB2/400 typically use the DRDA standard port number, 446, whereas DB2/UDB typically uses 50000 as the port number. Refer to IBM DB2 Administrator and Installation guides for locating and changing these port numbers for your DRDA server. For additional information, consult your DB2 DBA or System Administrator.

DRDA_DEFAULT_CCSID

Property	Description
Default Value	None
Range of Values	Any supported DRDA Server CCSID
Syntax	DRDA_DEFAULT_CCSID= <i>ccsid</i>

DRDA_DEFAULT_CCSID specifies the default CCSID or character set codepage for character set conversions when the DRDA Server database indicates that a character string has a CCSID of 65535. DRDA Servers use CCSID 65535 for columns specified as "FOR BIT DATA". In most cases, this parameter should not be specified, allowing CCSID 65535 to be treated as an Oracle RAW data type.

This parameter is for supporting databases (in particular, DB2/400) that use CCSID 65535 as the default for all tables created. Allowing CCSID 65535 to be treated as another CCSID can save such sites from having to modify every table.

Warning: Specifying any value for DRDA_DEFAULT_CCSID causes all "FOR BIT DATA" columns to be handled as text columns that need character set conversion and, therefore, any binary data in these columns can encounter conversion errors (ORA-28527).

DRDA_DESCRIBE_TABLE

Property	Description
Default Value	TRUE
Range of Values	{TRUE FALSE}
Syntax	DRDA_DESCRIBE_TABLE={ <i>TRUE</i> / <i>FALSE</i> }

DRDA_DESCRIBE_TABLE directs the gateway to use the DRDA operation `Table Describe` to return the description of tables. This is an optimization that reduces the amount of time and resources that are used to look up the definition of a table.

Note: This feature is not compatible with DB2 Aliases or Synonyms. If you use DB2 aliases, then be sure to disable this option.

DRDA_DISABLE_CALL

Property	Description
Default Value	TRUE
Range of Values	{TRUE FALSE}
Syntax	DRDA_DISABLE_CALL={ TRUE FALSE}

DRDA_DISABLE_CALL controls stored procedure usage, and is also used to control how the package is bound on the target database. This parameter should be set to FALSE only for supported target DRDA servers and should be set to TRUE otherwise.

Note: Any change to this parameter requires you to rebind.

DRDA_FLUSH_CACHE

Property	Description
Default Value	SESSION
Range of Values	{SESSION COMMIT}
Syntax	DRDA_FLUSH_CACHE={ SESSION COMMIT}

DRDA_FLUSH_CACHE specifies when the cursor cache is to be flushed. With DRDA_FLUSH_CACHE=COMMIT, the cursor cache is flushed whenever the transaction is committed. With DRDA_FLUSH_CACHE=SESSION, the cache is not flushed until the session terminates.

DRDA_GRAPHIC_CHAR_SIZE

Property	Description
Default Value	4
Range of Values	1 through 4
Syntax	DRDA_GRAPHIC_CHAR_SIZE=num

DRDA_GRAPHIC_CHAR_SIZE is used to define the character conversion size to be used for GRAPHIC data types. It is a tuning parameter which affects the maximum size of a GRAPHIC data type when the column is described.

DRDA_GRAPHIC_PAD_SIZE

Property	Description
Default Value	0
Range of Values	0 through 127
Syntax	DRDA_GRAPHIC_PAD_SIZE=num

DRDA_GRAPHIC_PAD_SIZE is used to pad the size of a Graphic column as described by the DRDA Server. This is sometimes necessary depending upon the character set of the DRDA database and Oracle database. If the Oracle database is based on EBCDIC and the DRDA database is based on ASCII, then a pad size of 2 may be needed.

DRDA_GRAPHIC_LIT_CHECK

Property	Description
Default Value	FALSE
Range of Values	{TRUE FALSE}
Syntax	DRDA_GRAPHIC_LIT_CHECK= { TRUE FALSE }

DRDA_GRAPHIC_LIT_CHECK directs the gateway to evaluate string literals within INSERT SQL statements in order to determine if they need to be converted to double-byte format for insertion into a Graphic column at the DRDA Server database. This is done by querying the column attributes of the table in the SQL statement to determine if a string literal is being applied to a column with a Graphic data type. If the table column is Graphic, and if this parameter is TRUE, then the gateway will rewrite the SQL statement with the literal converted to double-byte format. Existing double-byte characters in the string will be preserved, and all single-byte characters will be converted to double-byte characters.

DRDA_GRAPHIC_TO_MBCS

Property	Description
Default Value	FALSE
Range of Values	{TRUE FALSE}
Syntax	DRDA_GRAPHIC_TO_MBCS={ TRUE FALSE }

DRDA_GRAPHIC_TO_MBCS directs the gateway to convert graphic data that has been fetched from the DRDA Server into Oracle multi-byte data, translating double-byte characters into single-byte characters where possible.

DRDA_ISOLATION_LEVEL

Property	Description
Default Value	CHG for DB2/400, CS for DB2/OS390, DB2/UDB
Range of Values	{CHG CS RR ALL NC}

Property	Description
Syntax	DRDA_ISOLATION_LEVEL={ <i>CHG</i> / <i>CS</i> / <i>RR</i> / <i>ALL</i> / <i>NC</i> }

DRDA_ISOLATION_LEVEL specifies the isolation level that is defined to the package when it is created. All SQL statements that are sent to the remote DRDA database are executed with this isolation level. Isolation level seriously affects performance of applications. Use caution when specifying an isolation level other than the default. For information on isolation levels, refer to your IBM database manuals.

The following table lists the isolation levels and their descriptions. The levels are specified in ascending order of control, with CHG having the least reliable cursor stability and RR having the most. Note that higher stability uses more resources on the server and can lock those resources for extended periods.

Table C-1 Isolation Levels and Their Descriptions

Level	Description
CHG	Change (default for DB2/400)
CS	Cursor Stability (default for DB2/UDB, and DB2/OS390)
RR	Repeatable Read
ALL	ALL
NC	No Commit

Note: Any change to this parameter requires you to rebind.

DRDA_LOCAL_NODE_NAME

Property	Description
Default Value	AIX_RS6K
Range of Values	any alphanumeric string 1 to 8 characters in length
Syntax	DRDA_LOCAL_NODE_NAME= <i>name</i>

DRDA_LOCAL_NODE_NAME specifies the name by which the gateway will be known to the DRDA Server. This name is used internally by the DRDA Server to identify the local node.

DRDA_MBCS_TO_GRAPHIC

Property	Description
Default Value	FALSE
Range of Values	{TRUE FALSE}
Syntax	DRDA_MBCS_TO_GRAPHIC={ <i>TRUE</i> / <i>FALSE</i> }

DRDA_MBCS_TO_GRAPHIC directs the gateway to convert multi-byte data (that has been sent from Oracle to the DRDA database) into pure double-byte characters. This parameter is primarily intended to be used with bind variables in order to ensure that

the data is properly formatted and will therefore be acceptable to the DRDA Server. It applies only to INSERT SQL statements that are using bind variables. When used in combination with the `DRDA_GRAPHIC_LIT_CHECK` parameter, this parameter can help ensure that data that is being inserted into a Graphic column is handled correctly by the target DRDA Server.

DRDA_OPTIMIZE_QUERY

Property	Description
Default Value	TRUE
Range of Values	{TRUE FALSE}
Syntax	DRDA_OPTIMIZE_QUERY={ TRUE / FALSE }

`DRDA_OPTIMIZE_QUERY` enables or disables the distributed query optimizer (DQO) capability. The DQO capability is useful for optimizing queries that access large amount of data, but it can add overhead to small queries.

See Also: *Oracle Database Gateway for DRDA User's Guide* for more information.

This parameter is valid only if the DRDA Server is DB2/OS390. If the DRDA Server is DB2/400 or DB2/UDB, then you must set the value to `FALSE`.

DRDA_PACKAGE_COLLID

Property	Description
Default Value	ORACLE
Range of Values	An alphanumeric string 1 to 18 characters in length
Syntax	DRDA_PACKAGE_COLLID= <i>collection_id</i>

`DRDA_PACKAGE_COLLID` specifies the package collection ID. Note that in DB2/400, the collection ID is actually the name of an AS/400 library.

Note: Any change to this parameter requires you to rebind the package.

DRDA_PACKAGE_CONSTOKEN

Property	Description
Default Value	None, use the sample provided
Range of Values	A 16-digit hexadecimal number
Syntax	DRDA_PACKAGE_CONSTOKEN= <i>hexnum</i>

`DRDA_PACKAGE_CONSTOKEN` specifies the package consistency token. This is a 16-digit hexadecimal representation of an 8-byte token. Oracle recommends that you do not change the consistency token. The consistency token used at runtime must

match the one used when the package is bound. The value depends on the DRDA Server being used.

Note: Any change to this parameter requires you to rebind the package.

DRDA_PACKAGE_NAME

Property	Description
Default Value	G2DRSQL
Range of Values	An alphanumeric string 1 to 18 characters in length
Syntax	DRDA_PACKAGE_NAME= <i>name</i>

DRDA_PACKAGE_NAME specifies the package name. Note that the package is stored in the DRDA Server under this name as a SQL resource. Refer to the DRDA Server documentation for length limitations package names. Many typical implementations restrict the length to 8 characters.

Note: Any change to this parameter requires that you rebind the package.

DRDA_PACKAGE_OWNER

Property	Description
Default Value	None
Range of Values	Any valid user ID
Syntax	DRDA_PACKAGE_OWNER= <i>userid</i>

DRDA_PACKAGE_OWNER specifies the database user ID that owns the package. This enables the owner to be a user other than the connected user ID when the package is created. The package owner must be the same user as the owner of the ORACLE2PC table.

Note: Any change to this parameter requires you to rebind the package.

DRDA_PACKAGE_SECTIONS

Property	Description
Default Value	100
Range of Values	Any integer between 1 and 65535
Syntax	DRDA_PACKAGE_SECTIONS= <i>num</i>

DRDA_PACKAGE_SECTIONS specifies the number of cursors declared at the remote database when the package is bound. This is the maximum number of open cursors

permitted at any one time. Change this parameter only if an application needs more than 100 open concurrent cursors.

Note: Any change to this parameter requires you to rebind the package.

DRDA_READ_ONLY

Property	Description
Default Value	FALSE
Range of Values	{TRUE FALSE}
Syntax	DRDA_READ_ONLY={ TRUE FALSE}

DRDA_READ_ONLY specifies whether the gateway runs in a read-only transaction mode. In this mode, SQL statements that modify data are not permitted.

DRDA_RECOVERY_PASSWORD

Property	Description
Default Value	none
Range of Values	any valid password
Syntax	DRDA_RECOVERY_PASSWORD= <i>passwd</i>

DRDA_RECOVERY_PASSWORD is used with the DRDA_RECOVERY_USERID. The recovery user connects to the IBM database if a distributed transaction is in doubt.

See Also: *Oracle Database Gateway for DRDA User's Guide* for more information.

DRDA_RECOVERY_USERID

Property	Description
Default Value	ORARECOV
Range of Values	Any valid user ID
Syntax	DRDA_RECOVERY_USERID= <i>userid</i>

DRDA_RECOVERY_USERID specifies the user ID that is used by the gateway if a distributed transaction becomes in doubt. This user ID must have execute privileges on the package and must be defined to the IBM database.

If a distributed transaction becomes in doubt, then the Oracle database determines the status of the transaction by connecting to the IBM database, using the DRDA_RECOVERY_USERID. If this parameter is missing, then the gateway attempts to connect to a user ID of ORARECOV.

DRDA_REMOTE_DB_NAME

Property	Description
Default Value	DB2V2R3
Range of Values	An alphanumeric string 1 to 18 characters in length
Syntax	DRDA_REMOTE_DB_NAME= <i>name</i>

DRDA_REMOTE_DB_NAME specifies the DRDA Server location name. This is an identifying name that is assigned to the server for DRDA purposes. A technique for determining this name by using a SQL SELECT statement is discussed in each of the server-specific installation sections in [Chapter 13, "Configuring the DRDA Server"](#).

FDS_CLASS

Property	Description
Default Value	TG4DRDA_DB2MVS
Range of Values	Refer to the list below for valid values
Syntax	FDS_CLASS= <i>TG4DRDA_DB2MVS</i>

FDS_CLASS specifies the capability classification used by Oracle database and the gateway. These values may change from release to release, depending on whether the gateway capabilities change.

The valid default values for FDS_CLASS are as follows:

For a DB2/OS390 database: TG4DRDA_DB2MVS

For a DB2/400 database: TG4DRDA_DB2400

For a DB2/UDB database: TG4DRDA_DB2UDB

HS_NLS_NCHAR

Property	Description
Default Value	None
Range of Values	Any valid character set specification
Syntax	HS_NLS_NCHAR= <i>character_set</i>

HS_NLS_NCHAR specifies the character set that the gateway will use to interact with the DRDA Server when accessing Graphic data. Set this parameter to the same value as the character set component of the HS_LANGUAGE parameter. For additional details, refer to [Appendix E, "Globalization Support for DRDA"](#) and to the *Oracle Database Heterogeneous Connectivity Administrator's Guide*.

LOG_DESTINATION

Property	Description
Default Value	\$ORACLE_HOME/dg4drda/log/gateway sid_pid.log
Range of Values	Any valid file path
Syntax	LOG_DESTINATION= <i>logpath</i>

LOG_DESTINATION specifies the destination for gateway logging and tracing. This parameter should specify a file. If the file already exists, it will be overwritten.

After any failure to open the logpath, a second attempt to open the default is made.

Usually, LOG_DESTINATION should specify a directory. If it is specified as a file, and if two or more users simultaneously use the same instance of the gateway, then they are writing to the same log. The integrity of this log is not guaranteed. If you do not specify this parameter, then the default is assumed.

ORA_MAX_DATE

Property	Description
Default Value	4712-12-31
Range of Values	Any valid date less than 4712-12-31
Syntax	ORA_MAX_DATE= <i>yyyy-mm-dd</i>

ORA_MAX_DATE specifies the gateway maximum date value. If the fetched date value is larger than 4712-12-31, the gateway replaces the date value with the value defined by the ORA_MAX_DATE parameter. Any date between January 1, 4712 BC and December 31, 4712 AD is valid.

ORA_NLS11

Property	Description
Default Value	\$ORACLE_HOME/nls/data
Range of Values	Any valid Globalization Support directory path
Syntax	SET ORA_NLS11= <i>nlspath</i>

ORA_NLS11 specifies the directory to which the gateway loads its character sets and other language data. Normally this parameter does not need to be set. Some configurations, however, may require that it be set.

ORACLE_DRDA_TCTL

Property	Description
Default Value	None
Range of Values	Any valid file path

Property	Description
Syntax	ORACLE_DRDA_TCTL= <i>tracecontrolpath</i>

ORACLE_DRDA_TCTL specifies the path to the DRDA internal trace control file. This file contains module tracing commands. A sample file is stored in `$ORACLE_HOME/dg4drda/admin/debug.tctl`. This parameter is used for diagnostic purposes.

ORACLE_DRDA_TRACE

Property	Description
Default Value	value specified for LOG_DESTINATION
Range of Values	any valid file path
Syntax	ORACLE_DRDA_TRACE= <i>logpath</i>

ORACLE_DRDA_TRACE is used to specify a different log path for DRDA internal tracing. This tracing is separate from the rest of the gateway tracing, as specified by the LOG_DESTINATION parameter. By default, this parameter will append the DRDA internal trace to the gateway trace. This parameter is used for diagnostic purposes.

TRACE_LEVEL

Property	Description
Default Value	0
Range of Values	0-255
Syntax	TRACE_LEVEL= <i>number</i>

TRACE_LEVEL specifies a code tracing level. This value determines the level of detail which is logged to the gateway logfile during execution. This parameter is primarily used for diagnostics.

HS_NLS_DATE_FORMAT

Property	Description
Default value	Value determined by the HS_LANGUAGE initialization parameter
Range of values	Any valid date format mask (up to 255 characters)

Defines the date format for dates used by the target system. This initialization parameter has the same function as the NLS_DATE_FORMAT initialization parameter for an Oracle database. The value can be any valid date mask listed in the *Oracle Database SQL Language Reference*, but must match the date format of the target system. For example, if the target system stores the date February 14, 2001 as 2001/02/14, set the parameter to `yyyy/mm/dd`. Note that characters must be lowercase.

HS_NLS_DATE_LANGUAGE

Property	Description
Default value	Value determined by the HS_LANGUAGE initialization parameter
Range of values	Any valid NLS_LANGUAGE value (up to 255 characters)

Specifies the language used in character date values coming from the non-Oracle system. Date formats can be language independent. For example, if the format is `dd/mm/yyyy`, all three components of the character date are numeric. In the format `dd-mon-yyyy`, however, the month component is the name abbreviated to three characters. The abbreviation is language dependent. For example, the abbreviation for the month April is "apr", which in French is "avr" (Avril).

Heterogeneous Services assumes that character date values fetched from the non-Oracle system are in this format. Also, Heterogeneous Services sends character date bind values in this format to the non-Oracle system.

HS_NLS_NUMERIC_CHARACTER

Property	Description
Default value	Value determined by the HS_LANGUAGE initialization parameter
Range of values	Any valid NLS_NUMERIC_CHARACTERS value (any two valid numeric characters)

Specifies the characters to use as the group separator and the decimal character. The group separator separates integer groups (such as thousands, millions, and billions). The decimal character separates the integer portion of a number from the decimal portion.

D

Configuration Worksheet for DRDA

The table below is a worksheet that lists all of the parameter names and the reasons that you will need them for configuring the gateway and TCP/IP. Use the worksheet to gather the specific information that you need before you begin the configuration process.

Table D-1 List of Parameters Needed to Configure the Gateway

Reason	Name of Parameter Needed	Your Specific Parameters Here
Oracle home of the gateway	ORACLE_HOME	
System ID of the gateway	ORACLE_SID	
Remote collection ID	DRDA_PACKAGE_COLLID	
Remote package name	DRDA_PACKAGE_NAME	
Configuring TCP/IP	Local Hostname, Domain Name	
Configuring TCP/IP	IP Address	
Configuring TCP/IP	Network Mask	
Configuring TCP/IP	Name Server IP Address	
Configuring TCP/IP	DRDA Server Hostname or IP Address	
Configuring TCP/IP	DRDA Server Service Port Number	
Recovery user ID	DRDA_RECOVERY_USERID	
Recovery Password	DRDA_RECOVERY_PASSWORD	

Table D-1 (Cont.) List of Parameters Needed to Configure the Gateway

Reason	Name of Parameter Needed	Your Specific Parameters Here
Remote Database Name	DRDA_REMOTE_DB_NAME	
Connection Parameter	DRDA_CONNECT_PARM	
Owner ID of DRDA package	DRDA_PACKAGE_OWNER	
DB Name used with Oracle database	HS_DB_NAME	
DB Domain used with Oracle database	HS_DB_DOMAIN	

Note: The user ID that is used to bind or rebind the DRDA package must have the appropriate privileges on the remote database as described in [Chapter 13, "Configuring the DRDA Server"](#). Your database administrator will need to provide these privileges.

Globalization Support for DRDA

This appendix discusses the Globalization Support information for the Oracle Database Gateway for DRDA. This supplements the general Oracle Globalization Support information found in the *Oracle Database Advanced Application Developer's Guide*.

Globalization Support enables users to interact with Oracle applications in their native language, using their conventions for displaying data. The Oracle Globalization Support architecture is datadriven, enabling support for specific languages and character encoding schemes to be added without any changes in the source code.

There are a number of different settings in the gateway, DRDA Server, Oracle Database 11g, and client that affect Globalization Support processing. For translations to take place correctly, character settings of these components must be compatible.

This appendix contains the following sections:

- [Overview of Globalization Support Interactions](#)
- [Client and Oracle Database Configuration](#)
- [Gateway Language Interaction with DRDA Server](#)
- [Gateway Codepage Map Facility](#)
- [Multi-Byte and Double-Byte Support in the Gateway](#)
- [Message Availability](#)
- [Example of Globalization Support Configuration](#)

Overview of Globalization Support Interactions

Figure E-1 illustrates Globalization Support interactions within your computer, including each component of your computer and the parameters of each component that affect Globalization Support processing in a distributed environment.

Figure E-1 Architecture of Globalization Support Interactions with Your System Components

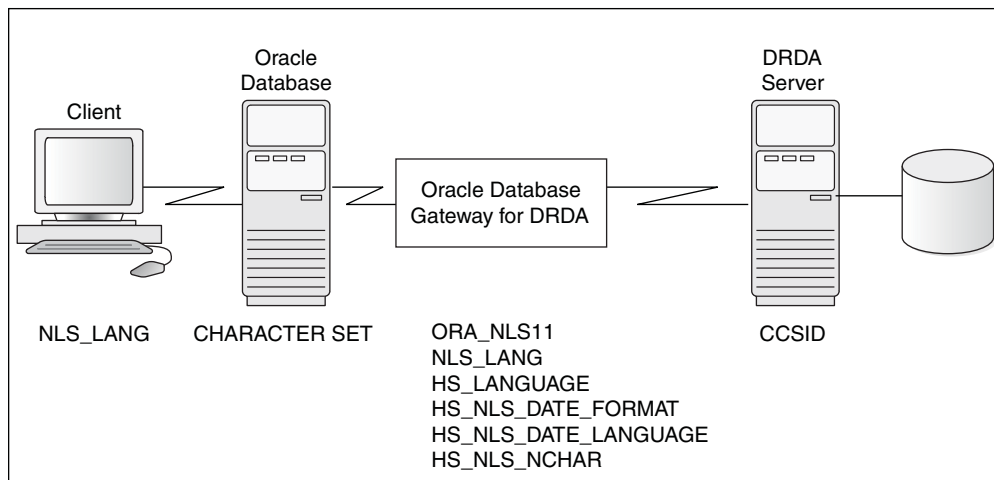


Table E-1 describes the architecture illustrated in Figure E-1.

The table discusses describes in detail the parameters and variables needed for Globalization Support processing within each of your computer’s environments: the client environment, the Oracle database, the gateway and the DRDA Server.

Parameters Needed for Globalization Support Processing in Your System’s Environments

Table E-1 Parameters Needed for Globalization Support Processing in Your Systems Environments

Environment	Parameter or Variable	Description
Client	NLS_LANG	It is environment variable. NLS_LANG sets the Globalization Support environment used by the database both for the server session and for the client application. This ensures that the language environments of both the database and client application are automatically the same. Because NLS_LANG is an environment variable, it is read by the client applications at startup. The client communicates the information defined in NLS_LANG to the server when it connects. For detailed information, refer to "Client and Oracle Database Configuration" on page E-3.
Oracle database	CHARACTER SET	This option is set during the creation of the database. CHARACTER SET determines the character encoding scheme used by the database and is defined at database creation in the CREATE DATABASE statement. All data columns of type CHAR, VARCHAR2, and LONG have their data stored in the database character set. For detailed information, refer to "Client and Oracle Database Configuration" on page E-3.
Oracle Database Gateway for DRDA	ORA_NLS11	It is environment variable. ORA_NLS11 determines where the gateway loads its character sets and other language data. For detailed information, refer to "Gateway Language Interaction with DRDA Server" on page E-4.
Oracle Database Gateway for DRDA	NLS_LANG	It is environment variable. NLS_LANG defines the character set used for communication between the gateway and the Oracle database. For detailed information, refer to "Gateway Language Interaction with DRDA Server" on page E-4.

Table E-1 (Cont.) Parameters Needed for Globalization Support Processing in Your Systems

Environment	Parameter or Variable	Description
Oracle Database Gateway for DRDA	HS_LANGUAGE	It is initialization parameter. HS_LANGUAGE defines the character set used for communications between the gateway and the DRDA Server. For detailed information, refer to " Gateway Language Interaction with DRDA Server " on page E-4.
Oracle Database Gateway for DRDA	HS_NLS_NCHAR	It is initialization parameter. HS_NLS_NCHAR defines the NCHAR character set that is used for communications between the gateway and the DRDA Server. This parameter is required when the gateway accesses GRAPHIC or multi-byte data on the DRDA Server. Set this parameter to the same value as the character set component of the HS_LANGUAGE parameter. For detailed information, refer to " Gateway Language Interaction with DRDA Server " on page E-4.
Oracle Database Gateway for DRDA	HS_NLS_DATE_FORMAT	It is initialization parameter. HS_NLS_DATE_FORMAT specifies the format for dates used by the DRDA Server. For detailed information, refer to " Gateway Language Interaction with DRDA Server " on page E-4.
Oracle Database Gateway for DRDA	HS_NLS_DATE_LANGUAGE	It is initialization parameter. HS_NLS_DATE_LANGUAGE specifies the language used by the DRDA Server for day and month names, and for date abbreviations. For detailed information, refer to " Gateway Language Interaction with DRDA Server " on page E-4.
DRDA Server	CCSID	CCSID is the server character set that is mapped in the gateway to the equivalent Oracle database character set. The CCSID specifies the character set that the DRDA database uses to store data. It is defined when you create a database. For detailed information, refer to " Gateway Codepage Map Facility " on page E-6.

Client and Oracle Database Configuration

A number of Globalization Support parameters control Globalization Support processing between the Oracle database and client. You can set language-dependent behavior defaults for the server, as well as for the client that overrides these defaults. For a complete description of Globalization Support parameters, refer to the Globalization Support chapter in the *Oracle Database Administrator's Guide*. These parameters do not directly affect gateway processing. However, you must ensure that the client character set (which is specified by the Oracle database NLS_LANG environment variable) is compatible with the character sets that you specify on the gateway and on the DRDA Server.

When you create an Oracle database, the character set that is used to store data is specified by the CHARACTER SET clause of the CREATE DATABASE statement. After the database is created, the database character set cannot be changed unless you re-create the database.

Normally, the default for CHARACTER SET is US7ASCII, which supports only the 26 Latin alphabetic characters. If you have specified 8-bit character sets on the gateway and the DRDA Server, then you must have a compatible 8-bit character set defined on your database. To check the character set of an existing database, issue the command:

```
SELECT USERENV('LANGUAGE') FROM DUAL;
```

For more information, refer to "Specifying Character Sets" in the *Oracle Database Administrator's Guide*.

Note that this does not mean that the gateway character set must be the same as the Oracle database character set. The Oracle Net facility performs implicit conversion between the Oracle database character set and the gateway character set.

Gateway Language Interaction with DRDA Server

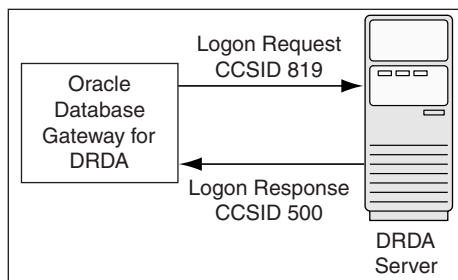
During logon of the gateway to the DRDA Server, initial language information is exchanged between the gateway and the server. First, the gateway sends to the DRDA Server the CCSID in which it will be conversing. In the following example, the Oracle database character set `WE8ISO8859P1` is mapped to CCSID 819 (an ASCII Code Page). This CCSID is sent to the DRDA Server. The DRDA Server responds with the CCSID in which it will be conversing. This will be the CCSID with which the DB2 database was generated. Also in the following example, this is CCSID 500, an EBCDIC Code Page. [Figure E-2, "Gateway Language Interaction with DRDA Server"](#), illustrates this process.

A DB2 instance maps unknown CCSIDs using the `SYSTEM.SYSSTRINGS` table. This table has different names for the various DB2 versions. It is possible to add additional character set mappings to this table by using DB2 utilities. Refer to the DB2 Installation documentation for details.

The setting of the `HS_LANGUAGE` parameter in the gateway `initsid.ora` determines which CCSID is used by the gateway for the conversation. Similarly, the setting of the `HS-NLS_NCHAR` parameter determines which CCSID is used by the gateway for GRAPHIC data interchange. For the list of supported ASCII-based Oracle database character sets that are mapped to CCSIDs, refer to ["Gateway Codepage Map Facility"](#) on page E-6.

Note that it is not necessary for the gateway character set to be the same as the Oracle database character set. In many cases, it is not feasible to set the gateway character set equal to the Oracle database character set because the DRDA Server does not have a valid translation for it. Instead, choose a character set that has the complete intersection with the character set that is used by the DRDA Server. The Oracle Net facility do as any translation between the gateway character set and the Oracle database character set.

Figure E-2 Gateway Language Interaction with DRDA Server



Gateway Configuration

After the gateway is installed, there are several parameters that you must change to customize for Globalization Support support.

Globalization Support Parameters in the Gateway Initialization File

There are four parameters in the Gateway Initialization File, `initsid.ora`, which affect Globalization Support:

- HS_LANGUAGE
- HS-NLS_NCHAR
- HS-NLS_DATE_FORMAT
- HS-NLS_DATE_LANGUAGE

HS_LANGUAGE

HS_LANGUAGE defines the character set that is used for communication between the gateway and the DRDA Server. It specifies the conventions, such as, the language used for messages from the target system, names of days and months, symbols for AD, BC, AM, and PM, and default language sorting mechanism.

The syntax of the HS_LANGUAGE parameter is:

```
HS_LANGUAGE=language[_territory.character_set]
```

where:

language can be any valid language.

territory is optional, and defaults to AMERICA.

character_set is optional and defaults to US7ASCII. This must be an ASCII base character set name, and it should match a character set listed in the gateway codepage map. Refer to "[Gateway Codepage Map Facility](#)" on page E-6 for the list of supplied character set mappings.

If you omit the HS_LANGUAGE parameter from `initsid.ora`, then the default setting is AMERICAN_AMERICA.US7ASCII. EBCDIC character sets are not supported. The values for *language* and *territory* (such as AMERICAN_AMERICA) must be valid, but they have no effect on translations.

HS-NLS_NCHAR

HS-NLS_NCHAR specifies the character set that is used by the gateway to interchange GRAPHIC data. For correct compatibility, set it to the same character set name that is specified in the HS_LANGUAGE parameter. If it is set to a character set other than that specified in HS_LANGUAGE, or if it is omitted, then translation errors will occur.

HS-NLS_DATE_FORMAT

HS-NLS_DATE_FORMAT specifies the format for dates used by the DRDA Server.

The syntax of the NLS_DATE_FORMAT parameter is:

```
HS-NLS_DATE_FORMAT=date_format
```

where *date_format* must be YYYY-MM-DD, the ISO date format. If this parameter is set to any other value or is omitted, then you receive an error when updating, deleting from, selecting from, or inserting into a table with date columns.

HS-NLS_DATE_LANGUAGE

HS-NLS_DATE_LANGUAGE specifies the language used by the DRDA Server for day and month names, and for date abbreviations. Because ISO date format contains numbers only, this parameter has no effect on gateway date processing and should be omitted.

Gateway Codepage Map Facility

The gateway has a user specifiable facility to map IBM Coded Character Set Identifiers (CCSIDs) to Oracle database character sets for the purpose of data translation.

The default map name is `codepage.map` and is located in the `$ORACLE_HOME/dg4drda/admin` directory. Refer to [Appendix C, "Initialization Parameters"](#) for more detailed information about the `DRDA_CODEPAGE_MAP` parameter.

The map has two different forms of syntax. The first form of syntax defines a mapping between a CCSID and an Oracle database character set:

```
[S|D|M] CCSID direction Oracle_CharacterSet {shift}
```

Where:

S designates a single-byte character set

D designates a double-byte character set

M designates a multi-byte character set

CCSID is the IBM coded character set identifier

direction is one of the following:

- = mapping is bidirectional
- < mapping is one-way, Oracle database character set to CCSID
- > mapping is one-way, CCSID to Oracle database character set

Oracle_CharacterSet is the name of a valid Oracle database character set.

shift indicates a character set that requires Shift OUT/IN processing. Set this attribute only for EBCDIC-based double-byte and multi-byte mappings.

The second form of syntax defines a mapping of a multi-byte CCSID to its single-byte and double-byte CCSID equivalents:

```
MBC multi = single double
```

Where:

multi is the multi-byte CCSID

single is the single-byte CCSID

double is the double-byte CCSID

This facility is intended for mapping CCSIDs which were not previously mapped as shipped with the gateway. You must contact Oracle Support Services before modifying this map.

The following are the contents of the map as shipped with the Oracle Database Gateway for DRDA:

```
# Copyright (c) 2001, 2007, Oracle Corporation. All rights reserved.
# Database gateway for DRDA - CodePage/Oracle CharacterSet Map
# S==Single-byte, D==Double-byte, M==Multi-byte, MBC==SBC DBC mapping
#
# Single-byte codepage mappings
#
S 37 = WE8EBCDIC37 # United States/Canada EBCDIC
S 273 = D8EBCDIC273 # Austria/Germany EBCDIC
S 277 = DK8EBCDIC277 # Denmark/Norway EBCDIC
```

```

S 278 = S8EBCDIC278 # Finland/Sweden EBCDIC
S 280 = I8EBCDIC280 # Italy EBCDIC
S 284 = WE8EBCDIC284 # Latin America/Spain EBCDIC
S 285 = WE8EBCDIC285 # United Kingdom EBCDIC
S 297 = F8EBCDIC297 # France EBCDIC
#S 420 = AR8EBCDICX # Arabic Bilingual (USA English) EBCDIC
S 420 = AR8XBASIC # Arabic Bilingual (USA English) EBCDIC
S 424 = IW8EBCDIC424 # Israel (Hebrew) EBCDIC
S 437 = US8PC437 # Personal Computer,USA ASCII
S 500 = WE8EBCDIC500 # International EBCDIC
S 813 = EL8ISO8859P7 # Greek ASCII
S 819 = WE8ISO8859P1 # ISO/ANSI Multilingual ASCII
S 838 = TH8TI8EBCDIC # Thai w/Low-Tone Marks & Ancient Chars EBCDIC
S 850 < US7ASCII # Multilingual Page - Personal Computer ASCII
S 850 = WE8PC850 # Multilingual Page - Personal Computer ASCII
S 864 = AR8ISO8859P6 # Arabic - Personal Computer ASCII
S 870 = EE8EBCDIC870 # Latin 2, Multilingual/ROECE EBCDIC
S 871 = WE8EBCDIC871 # Iceland - CECP EBCDIC
S 875 = EL8EBCDIC875 # Greece EBCDIC
S 904 > US7ASCII # Traditional Chinese - PC-Data ASCII
S 912 = EE8ISO8859P2 # Latin 2 8-bit ASCII
S 916 = IW8ISO8859P8 # Israel (Hebrew) ASCII
S 1025 = CL8EBCDIC1025 # Cyrillic, Multiling EBCDIC
S 1086 = IW8EBCDIC1086 # Israel EBCDIC
S 1252 = WE8MSWIN1252 # Latin 1 - MS-Windows ASCII
S 1253 = EL8MSWIN1253 # Greek - MS-Windows ASCII
S 28709 > WE8EBCDIC37 # United States/Canada (CP28709==CP37) EBCDIC
#
# Multi-byte codepage mappings
#
#S 833 > KO16DBCS # Korean Extended single-byte EBCDIC
#D 834 > KO16DBCS shift # Korean double-byte EBCDIC
#M 933 = KO16DBCS shift # Korean Mixed multi-byte EBCDIC

#MBC 933 = 833 834 # Korean Mixed multi-byte EBCDIC
#
#S 1088 > KO16MSWIN949 # Korean KS single-byte PC-Data ASCII
#D 951 > KO16MSWIN949 # Korean KS double-byte PC-Data ASCII
#M 949 = KO16MSWIN949 # Korean KS multi-byte PC-Data ASCII
#MBC 949 = 1088 951 # Korean KS multi-byte PC-Data ASCII
#
#S 891 > KO16KSC5601 # Korean single-byte ASCII
#S 1040 > KO16KSC5601 # Korean single-byte ASCII
#D 926 > KO16KSC5601 # Korean double-byte ASCII
#M 934 = KO16KSC5601 # Korean multi-byte ASCII
#M 944 > KO16KSC5601 # Korean multi-byte ASCII
#MBC 934 = 891 926 # Korean multi-byte ASCII
#MBC 944 = 1040 926 # Korean multi-byte Extended ASCII
#
#S 28709 > ZHT16DBCS # Traditional Chinese single-byte EBCDIC
#D 835 > ZHT16DBCS shift # Traditional Chinese double-byte EBCDIC
#M 937 = ZHT16DBCS shift # Traditional Chinese multi-byte EBCDIC
#MBC 937 = 28709 835 # Traditional Chinese multi-byte EBCDIC
#
#S 1114 > ZHT16MSWIN950 # Traditional Chinese single-byte ASCII
#D 947 > ZHT16MSWIN950 # Traditional Chinese double-byte ASCII
#M 950 = ZHT16MSWIN950 # Traditional Chinese multi-byte ASCII
#MBC 950 = 1114 947 # Traditional Chinese multi-byte ASCII
#

```

```

#S  836 > ZHS16DBCS      # Simplified Chinese single-byte      EBCDIC
#D  837 > ZHS16DBCS shift # Simplified Chinese double-byte     EBCDIC
#M  935 = ZHS16DBCS shift # Simplified Chinese multi-byte    EBCDIC
#MBC 935 = 836 837      # Simplified Chinese multi-byte    EBCDIC
#
#S 1027 > JA16DBCS      # Japanese single-byte      EBCDIC
#D  300 > JA16DBCS shift # Japanese double-byte     EBCDIC
#D 4396 > JA16DBCS shift # Japanese double-byte     EBCDIC
#M  939 = JA16DBCS shift # Japanese multi-byte      EBCDIC
#M 5035 > JA16DBCS shift # Japanese multi-byte      EBCDIC
#MBC 939 = 1027 300    # Japanese multi-byte      EBCDIC
#MBC 5035 = 1027 4396  # Japanese multi-byte      EBCDIC
#
#S  290 > JA16EBCDIC930 # Japanese single-byte      EBCDIC
#D  300 > JA16EBCDIC930 shift # Japanese double-byte     EBCDIC
#D 4396 > JA16EBCDIC930 shift # Japanese double-byte     EBCDIC
#M  930 = JA16EBCDIC930 shift # Japanese multi-byte      EBCDIC
#M 5026 > JA16EBCDIC930 shift # Japanese multi-byte      EBCDIC
#MBC 930 = 290 300    # Japanese multi-byte      EBCDIC
#MBC 5026 = 290 4396  # Japanese multi-byte      EBCDIC
#

```

Refer to the following list to check the character set of an existing database:

- **For DB2/OS390:** Ask your system administrator. There is no single command you use.
- **For DB2/400:** Issue the command `DSPSYSVAL SYSVAL (QCCSID)`
- **For DB2/UDB:** Ask your system administrator. There is no single command you use.

Multi-Byte and Double-Byte Support in the Gateway

To enable the gateway to properly handle double-byte and multi-byte data, you must configure the codepage map facility with proper multi-byte maps, and (as an option) can set the following gateway configuration parameters:

- `DRDA_GRAPHIC_LIT_CHECK`
- `DRDA_GRAPHIC_TO_MBCS`
- `DRDA_MBCS_TO_GRAPHIC`
- `DRDA_GRAPHIC_PAD_SIZE`
- `DRDA_GRAPHIC_CHAR_SIZE`

Refer to [Appendix C, "Initialization Parameters"](#), for the values of these parameters.

Configuring the codepage map requires knowledge of the codepages that have been configured in the DRDA Server database as well as knowledge of compatible Oracle database character sets.

IBM coded character set identifiers (CCSIDs) are used to indicate which codepage is configured as the primary codepage for the database, as well as any translation character sets loaded into the database. Some DRDA Servers, such as with DB2, have a translation facility in which character set transforms are mapped between two compatible character sets. For DB2/OS390, these transforms are stored in the table `SYSIBM.SYSSTRINGS` and transform on the CCSID codepage to another CCSID codepage. In `SYSSTRINGS`, `IN` and `OUT` columns specify the CCSIDs that are used in

the transform. Typical transforms are from ASCII to EBCDIC and back. Two transforms are therefore used for two given CCSIDs.

Multi-byte codepages comprise a single-byte codepage and a double-byte codepage. For example, the Korean EBCDIC multi-byte codepage, CCSID 933, is composed of two codepages: `codepage 833` (for single-byte) and `codepage 834` (for double-byte). The DRDA Server, therefore, can send data to the gateway in any of these three codepages, and the gateway must translate appropriately depending upon which codepage the data is associated with. Because CCSID 933 is an EBCDIC-based codepage, and the gateway must use an ASCII-based codepage, we identify an equivalent set of codepages, which are ASCII-based. For example the Korean multi-byte codepage, CCSID 949, which is composed of two codepages: `codepage 1088` (for single-byte) and `codepage 951` (for double-byte).

The codepage map facility is used to map these CCSIDs into the equivalent Oracle database character sets. Unlike IBM CCSIDs, Oracle database character sets are unified (in that single-byte and double-byte character sets have been combined into one set) and are identified by one ID instead of three IDs. In our previous example, the equivalent Oracle database character set for the ASCII Korean codepages would be `KO16MSWIN949`, and the EBCDIC Korean codepages would be `KO16DBCS`. These are identified to the gateway by using a set of mapping entries in the `codepage.map` file.

The EBCDIC Korean sets are:

```
S  833 > KO16DBCS      # Korean Extended single-byte      EBCDIC
D  834 > KO16DBCS shift # Korean double-byte            EBCDIC
M  933 = KO16DBCS shift # Korean Mixed multi-byte       EBCDIC
MBC 933 = 833 834     # Korean Mixed multi-byte       EBCDIC
```

Notice that the multi-byte set is a bidirectional map to `KO16DBCS`. However the single and double codepages are mapped one-way to `KO16DBCS`. Because only one bidirectional CCSID to Oracle database character set entry for a given pair can exist, you directly map the multi-byte sets. Because the single-byte and double-byte CCSIDs are ostensibly subsets of `KO16DBCS`, you map them as one-way entries. Note that double-byte and multi-byte maps are tagged with the `shift` attribute. This is required for EBCDIC double-byte and multi-byte codepages as part of the shift out/in encapsulation of data. The single-byte map is not marked because single-byte sets are not permitted to contain double-byte data and thus will never use shift encapsulation. Note that the MBC entry ties the codepages together.

The ASCII Korean sets are similarly mapped and are:

```
S  1088 > KO16MSWIN949 # Korean KS single-byte PC-Data    ASCII
D   951 > KO16MSWIN949 # Korean KS double-byte PC-Data   ASCII
M   949 = KO16MSWIN949 # Korean KS multi-byte PC-Data   ASCII
MBC 949 = 1088 951     # Korean KS multi-byte PC-Data   ASCII
```

Notice that the multi-byte set is a bidirectional map to `KO16MSWIN949`. However the single and double codepages are mapped one-way to `KO16MSWIN949`. Because only one bidirectional CCSID to Oracle database character set entry for a given pair can exist, we directly map the multi-byte sets. And because the single-byte and double-byte CCSIDs are ostensibly subsets of `KO16MSWIN949`, we map them as one-way entries. Note that there is no "shift" attribute in any of these mappings. This is because ASCII-based sets do not use shift out/in encapsulation. Instead, ASCII-based sets use a different method (which does not use a shift out/in protocol) to identify double-byte characters.

The above entries supply the necessary codepage mappings for the gateway. To complete the example, we need to specify the correct character set in the `HS_`

LANGUAGE and HS_NLS_NCHAR parameters in the gateway initialization file. The Gateway Initialization parameters would look as follows:

```
HS_LANGUAGE=AMERICAN_AMERICA.KO16MSWIN949
HS_NLS_NCHAR=KO16MSWIN949
```

Note that the specified character set must be ASCII-based.

This takes care of the configuration of the gateway. The last step is to set up transforms between the EBCDIC codepages and the ASCII codepages in the DRDA Server database. Normally, the gateway would use a total of six transforms, one of each pair in both directions. You may save some table space by installing only the ASCII-to-EBCDIC transforms, because the DRDA Server needs to translate only the ASCII data that is sent by the gateway, but the DRDA Server does not need to send ASCII data. The gateway will receive the EBCDIC data and translate as needed. This one-sided data transfer methodology is called receiver-makes-right, meaning that the receiver must translate whatever character set the sender uses. In our example, the DRDA Server is EBCDIC-based, so it will send all data in EBCDIC. The server, therefore, does not need to have an EBCDIC-to-ASCII transform because the server never use the transform.

In our previous example, the DRDA Server database is assumed to be EBCDIC, which is likely to be true for a DB2/OS390 database. For a DB2/UDB database, however, this is not likely to be true. Because most DB2/UDB databases run on ASCII-based computers, they will likely be created with ASCII-based codepages. In such cases, the gateway needs to have only one set of codepage map definitions, which are for the ASCII set. Also, because both the DRDA Server and the gateway will be using the same codepages, no character set transforms need to be loaded into the DB2 database. This can help reduce the amount of CPU overhead that is associated with character translation.

Be aware that some multi-byte codepages may be composed of single-byte CCSIDs that are already defined in the `codepage.map` file that is provided with the product. If you are adding a new set of entries to support a multi-byte set, then comment out the provided entries so that your new entries will be used correctly.

Additional codepage mappings, which are not already provided, are possible. You may construct entries such as those in our examples, given knowledge of the IBM CCSIDs and the Oracle database character sets. Because this can be complex and quite confusing (given the IBM documentation of codepage definitions and Oracle database character set definitions), please thoroughly test your definitions for all desired character data values before putting them into production.

If you are uncertain, then contact Oracle Support Services to request proper codepage mapping entries.

Message Availability

Whether a language message module is available depends on which modules are installed in the Oracle product set that is running on the server. If message modules for a particular language set are not installed, then specifying that language with a language parameter does not display messages in the requested language.

Example of Globalization Support Configuration

Following is an example of all the settings that are needed to configure the gateway, DRDA Server, Oracle database, and client so that a language and character set are

working compatibly across the system. In this example, the settings enable a customer in Germany to interact with the gateway in German:

gateway initsid.ora file:

```
HS_LANGUAGE=AMERICAN_AMERICA.WE8ISO8859P1
HS_NLS_DATE_FORMAT=YYYY-MM-DD
```

DRDA Server CCSID:

```
273 (D8EBCDIC273)
```

Oracle Database and Client Setting for Database:

```
SELECT USERENV('language') FROM DUAL;
USERENV('LANGUAGE')
```

```
-----
AMERICAN_AMERICA.WE8ISO8859P1
```

Oracle Database and Client Environment Variables:

```
NLS_LANG=GERMAN_GERMANY.WE8ISO8859P1
```

Index

A

AIX

AIX_RS6K, default value for DRDA_LOCAL_
NODE_NAME, C-21

alias

DB2, C-19

application

authenticating logons, 15-1

AS/400

command, DSPRDBDIRE, 13-3

library name, DRDA_PACKAGE_COLLID, C-22

ASCII

code page, E-4

US7ASCII, E-5

US7ASCII as default, E-3

B

Bind Package Stored Procedure

DB2/400, 13-3

DB2/OS390, 13-1

DB2/UDB, 13-3

BIND privilege

DB2/OS390, 13-1

DB2/UDB, 13-4

BINDADD authority, 14-6

BINDADD privilege

DB2/OS390, 13-2

DB2/UDB, 13-4

BINDAGENT privilege

DB2/OS390, 13-2

binding the DRDA package

configuring the gateway on DRDA server, 14-6

DB2/400, 13-3

DB2/OS390, 13-1

DB2/UDB, 13-3

C

CCSID

65535 as the default for all tables created, C-18

codepage mapping facility, E-6

description, E-3

CCSID (coded character set identifiers), defined, E-8

CHARACTER SET

clause, client/server configuration, E-3

parameter, description, E-2

character sets

and codepage map facility, E-6

CCSID, E-3

codepage, C-18

Heterogeneous Services, C-9

clause

CHARACTER SET, client/server

configuration, E-3

code tracing, C-27

codepage map facility

for data translation, E-6

collection privilege - CREATE IN

DB2/OS390, 13-2

collection privilege - CREATETAB

DB2/OS390, 13-2

Communication Database (CDB) tables, DDF, 13-2

concurrent connections

TCP/IP, 12-2

configuration assistants

troubleshooting, B-2

Configuring

two-phase commit, 3-8, 5-8, 7-8, 9-8

configuring

DB2/400, 13-3

DB2/OS390, 13-1

host workstation for gateway, 14-2

Configuring the gateway, 3-1, 5-1, 7-1, 9-1

CONNECT authority

DB2/UDB, 14-6

CONNECT privilege

DB2/UDB, 13-4

COPY privilege

DB2/OS390, 13-1

CREATE DATABASE statement, client/server

configuration, E-3

CREATE IN privilege

DB2/OS390, 13-2

CREATE PUBLIC DATABASE LINK privilege, 14-7

CREATEIN privilege

DB2/UDB, 13-4

CREATETAB authority, 14-6

CREATETAB privilege

DB2/OS390, 13-2

DB2/UDB, 13-4

Creating
transaction log table, 3-9, 5-9, 7-9, 9-9
cursor
number of cursors, DRDA_PACKAGE_ SECTIONS, C-23
stability, DRDA_ISOLATION_LEVEL, C-21

D

data dictionary
support, 14-1
database
authorities - CONNECT, BINDADD, and CREATETAB, 13-4
link
binding the gateway package, 14-7
defining and controlling, 15-2
Database link
behavior, 11-7
database link
behavior, 3-7, 5-7, 7-7, 9-7, 14-10
dropping, when deinstalling the gateway, 17-2
datatypes
RAW
DRDA_DEFAULT_CC SID, C-18
date formats
Heterogeneous Services, C-27
DB2
aliases, C-19
Distributed Data Facility (DDF), 13-2
DRDA_DESCRIBE_TABLE compatibility, C-19
SPUFI utility, 13-2
DB2/400
configuring, 13-3
DRDA_DEFAULT_CC SID, C-18
DRDA_ISOLATION_LEVEL, C-21
DRDA_OPTIMIZE_QUERY, C-22
DRDA_PACKAGE_COLLID, C-22
user ID mapping, 15-3
DB2/OS390
configuring, 13-1
DRDA_ISOLATION_LEVEL, C-21
DRDA_OPTIMIZE_QUERY, C-22
user ID mapping, 15-3
DB2/UDB
configuring, 13-3
DRDA_ISOLATION_LEVEL, C-21
DRDA_OPTIMIZE_QUERY, C-22
user ID mapping, 15-4
DDF
DB2 (Distributed Data Facility), 13-2
describe cache high water mark
definition, C-9
Destination Hostname or IP Address, same as DRDA server Hostname or IP Address (configuring TCP/IP, worksheet), D-1
Destination Service Port Number, same as DRDA Server Service Port Number (configuring TCP/IP, worksheet), D-1
diagnostic parameter, C-27

distributed
data facility (DDF), 13-2
distributed query optimizer (DQO)
DRDA-specific parameters, C-22
operations, DB2, 13-2
transaction, DRDA_RECOVERY_USERID, C-24
double-byte support, E-8
DQO
DRDA-specific parameters, C-22
DRDA
session security options, 15-3
DRDA Server
languages and character sets in configuration, E-6
DRDA server
configuring
DB2/400, 13-3
DB2/OS390, 13-1
DB2/UDB, 13-3
Hostname or IP Address (configuring TCP/IP, worksheet), D-1
Service Port Number (configuring TCP/IP, worksheet), D-1
DRDA_CODEPAGE_MAP parameter, E-6
DRDA_COMM_BUFL EN parameter, C-17
DRDA_CONNECT_PARM (TCP/IP format) parameter, C-17
DRDA_DEFAULT_CC SID parameter, C-18
DRDA_DESCRIBE_TABLE parameter defined, C-18
DRDA_DISABLE_CALL parameter defined, C-19
DRDA_FLUSH_CACHE parameter, C-19
DRDA_GRAPHIC_LIT_CHECK parameter, C-20
DRDA_GRAPHIC_PAD_SIZE parameter, C-20
DRDA_GRAPHIC_TO_MBCS parameter, C-20
DRDA_ISOLATION_LEVEL parameter, C-20
DRDA_LOCAL_NODE_NAME parameter, C-21
DRDA_MBCS_TO_GRAPHIC parameter, C-21
DRDA_OPTIMIZE_QUERY parameter defined, C-22
DRDA_PACKAGE_COLLID parameter defined, C-22
DRDA_PACKAGE_CONSTOKEN parameter, C-22
DRDA_PACKAGE_NAME parameter defined, C-23
DRDA_PACKAGE_OWNER parameter, C-23
DRDA_PACKAGE_SECTIONS parameter defined, C-23
DRDA_READ_ONLY parameter defined, C-24
DRDA_RECOVER_USERID
DB2/400, 13-3
DB2/OS390, 13-2
if the user ID is not specified, 13-4
DRDA_RECOVERY_PASSWORD parameter
DB2/400, 13-3
DB2/OS390, 13-2
defined, C-24
defining the recovery user ID, 13-4

DRDA_RECOVERY_USERID parameter
DB2/400, 13-3
DB2/OS390, 13-2
defined, C-24
defining the recovery user ID, 13-4
DRDA_REMOTE_DB_NAME parameter, C-25
DSPRDBDIRE command, 13-3

E

EBCDIC
character set support, E-5
code page, E-4
DRDA Server CCSID, E-11
environment variable
NLS_LANG
client character set, E-3
description of use, E-2
ORA_NLS11, E-2
error
date, E-5
obsolete parameters, 16-2
while binding the gateway package, 14-7
Error messages
error tracing, C-15
errors
configuration assistants, B-2
installation, B-2, B-3
non-interactive installation, B-3
silent mode, B-3
X windows, B-1
EXECUTE privilege
DB2/OS390, 13-1
DB2/UDB, 13-4

F

fatal errors, B-3
FDS_CLASS parameter, C-25
fetch array size, with HS_FDS_FETCH_ROWS, C-16
fetched date, C-26
file
InstallActions.log, 17-2
listener.ora, 17-2
tnsnames.ora, 17-2
files
oraInst.loc, A-2
response files, A-3
FOR BIT DATA
DRDA_DEFAULT_CCSD, C-18

G

Gateway
default SID, 3-1, 5-1, 7-1, 9-1
system identifier (SID), 3-1, 5-1, 7-1, 9-1, 11-1
two-phase commit, 3-8, 5-8, 7-8, 9-8
gateway
authenticating logons, 15-1
binding DRDA packages, 14-6
logging, LOG_DESTINATION, C-26

supported languages, E-6
tracing
LOG_DESTINATION, C-26
gateway initialization file
if errors are reported, 14-7
Gateway Password Encryption Tool, 3-9, 3-10, 5-10,
7-9, 9-10
Globalization Support
DRDA Server character sets, E-6
initsid.ora parameters, E-4
overview, E-1
parameters
configuration on client and Oracle
databases, E-3
globalization support
Heterogeneous Services, C-9
date format, C-27
languages in character date values, C-28
GTW\$_BIND_PKG
stored procedure, 14-7

H

hardware requirements, 12-1
Heterogeneous Services
defining maximum number of open cursors, C-10
initialization parameters, 11-1
optimizing data transfer, C-10
Oracle Database Gateway for ODBC
creating initialization file, 11-1
setting global name, C-8
specifying cache high water mark, C-9
tuning internal data buffering, C-11
tuning LONG data transfer, C-10
HS_CALL_NAME initialization parameter, C-7
HS_DB_NAME initialization parameter, C-8
HS_DESCRIBE_CACHE_HWM initialization
parameter, C-9
HS_FDS_CONNECT_INFO, C-12
HS_FDS_CONNECT_INFO initialization parameter
specifying connection information, 11-2
HS_FDS_DEFAULT_OWNER initialization
parameter, C-13
HS_FDS_FETCH_ROWS parameter, C-16
HS_FDS_PROC_IS_FUNC initialization
parameter, C-14
HS_FDS_RECOVERY_PWD initialization
parameter, C-16
HS_FDS_RESULTSET_SUPPORT initialization
parameter, C-15
HS_FDS_SHAREABLE_NAME initialization
parameter, C-16
HS_FDS_TRACE_LEVEL initialization
parameter, C-15
enabling agent tracing, C-2
HS_FDS_TRANSACTION_LOG initialization
parameter, C-15
HS_LANGUAGE initialization parameter, C-9
HS_LONG_PIECE_TRANSFER_SIZE initialization
parameter, C-10

HS_NLS_DATE_FORMAT
for dates used by the DRDA Server, E-5
HS_NLS_DATE_FORMAT initialization
parameter, C-27
HS_NLS_DATE_LANGUAGE, E-5
HS_NLS_DATE_LANGUAGE initialization
parameter, C-28
HS_NLS_NCHAR
character set to interchange GRAPHIC data, E-5
defined, C-25
HS_NLS_NUMERIC_CHARACTER initialization
parameter, C-28
HS_OPEN_CURSORS initialization parameter, C-10
HS_RPC_FETCH_REBLOCKING initialization
parameter, C-10
HS_RPC_FETCH_SIZE initialization
parameter, C-11
HS_TIME_ZONE initialization parameter, C-11
HS_TRANSACTION_LOG, 3-9, 5-9, 7-9, 9-9

I

IFILE initialization parameter, C-12
IN and OUT columns, multi-byte support, E-8
inbound connections
processing, 15-2
Initialization parameter file
customizing, 3-2, 5-1, 7-1, 9-2, C-1
initialization parameters
Heterogeneous Services (HS), 11-1
initialization parameters (HS)
Oracle Database Gateway for ODBC, 11-1
initsid.ora file, 3-2, 5-2, 7-2, 9-2
Globalization Support parameters, E-4
InstallActions.log file, 17-2
installation
errors, B-2, B-3
silent mode, B-3
log files, B-1
non-interactive
error handling, B-3
oraInst.loc file, A-2
response files, A-3
preparing, A-3
silent mode, B-3
templates, A-3
silent mode, A-4, A-5
internal tracing, C-27
IP name, 15-3
isolation level, DRDA_ISOLATION_LEVEL, C-21

L

LANGUAGE parameter, E-5
listener.ora file, 3-11, 5-11, 7-11, 9-11, 11-9, 14-11,
17-2
example, 3-3, 5-3, 7-3, 9-3, 11-4, 14-3
location, 3-2, 5-2, 7-2, 9-3, 11-3, 14-3
log files, B-1
troubleshooting, B-2

LOG_DESTINATION parameter
defined, C-26
with ORACLE_DRDA_TRACE, C-27
logging, LOG_DESTINATION, C-26

M

mount point directories, 1-5
multi-byte support, E-8

N

Net Configuration Assistant
troubleshooting, B-2
NLS_LANG
environmental variable, E-2
server-side parameter, E-2
NLS_LANG environment variable, E-3
non-interactive installation
oraInst.loc file, A-2
response files
preparing, A-3
templates, A-3
silent mode, A-4, A-5
errors, B-3
non-interactive installations
running
Oracle Universal Installer, A-5
number of cursors, DRDA_PACKAGE_
SECTIONS, C-23

O

obsolete parameters since V4 gateway, 16-2
ODBC connectivity
specifying path to library, C-16
operating system
user ID for DB2/UDB, 13-4
option
binding packages, 14-6
DRDA session security, 15-3
security conduct, 15-2
service port number, DRDA_CONNECT_
PARAM, C-18
ORA_MAX_DATE parameter, C-26
ORA_NLS11 parameter, C-26
description, E-2
Oracle
RAW datatype, C-18
Oracle Database Gateway for ODBC
creating initialization file, 11-1
Oracle Net
configuring, 3-2, 5-2, 7-2, 9-2, 11-3, 14-3
operating system authentication, 15-2
Oracle Net Listener
starting, 3-5, 11-5
Oracle Universal Installer
using
to de-install the gateway, 17-1
ORACLE_DRDA_TCTL parameter
defined, C-26

ORACLE_DRDA_TRACE parameter, defined, C-27
ORACLE2PC table
DB2/400, 13-3
DB2/OS390, 13-1
DB2/UIDB, 13-4
DRDA_PACKAGE_OWNER, C-23
ORARECOV user ID
DB2/400, 13-3
DB2/OS390, 13-2
DB2/UIDB, 13-4
DRDA_RECOVERY_USERID, C-24

P

package
collection id, DRDA_PACKAGE_COLLID, C-22
consistency token, DRDA_PACKAGE_
CONSTOKEN, C-22
privileges - BIND and EXECUTE,
DB2/UIDB, 13-4
privileges - BIND, COPY, and EXECUTE
DB2/OS390, 13-1
parameter
diagnostic, C-27
obsolete since V4 gateway, 16-2
parameter file
tailoring to configure the host, 14-2
parameters
DRDA_CODEPAGE_MAP
defined, C-17
mapping IBM CCSID, E-6
DRDA_RECOVERY_PASSWORD
DB2/400, 13-3
DB2/OS390, 13-2
DB2/UIDB, 13-4
DRDA_RECOVERY_USERID
DB2/400, 13-3
DB2/OS390, 13-2
DB2/UIDB, 13-4
FDS_CLASS, C-25
gateway initialization file
DRDA_CACHE_TABLE_DESC, C-16
DRDA_CAPABILITY, C-17
DRDA_CODEPAGE_MAP, C-17
DRDA_COMM_BUFLN, C-17
DRDA_CONNECT_PARM (TCP/IP
format), C-17
DRDA_DEFAULT_CCSID, C-18
DRDA_DESCRIBE_TABLE, C-18
DRDA_DISABLE_CALL, C-19
DRDA_FLUSH_CACHE, C-19
DRDA_GRAPHIC_LIT_CHECK, C-20
DRDA_GRAPHIC_PAD_SIZE, C-20
DRDA_GRAPHIC_TO_MBCS, C-20
DRDA_ISOLATION_LEVEL, C-20
DRDA_LOCAL_NODE_NAME, C-21
DRDA_MBCS_TO_GRAPHIC, C-21
DRDA_OPTIMIZE_QUERY, C-22
DRDA_PACKAGE_COLLID, C-22
DRDA_PACKAGE_CONSTOKEN, C-22

DRDA_PACKAGE_NAME, C-23
DRDA_PACKAGE_OWNER, C-23
DRDA_PACKAGE_SECTIONS, C-23
DRDA_READ_ONLY, C-24
DRDA_RECOVERY_PASSWORD, C-24
DRDA_RECOVERY_USERID, C-24
DRDA_REMOTE_DB_NAME, C-25
HS_FDS_FETCH_ROWS, C-16
HS-NLS_NCHAR, C-25
LOG_DESTINATION, C-26
ORA_MAX_DATE, C-26
ORA-NLS11, C-26
ORACLE_DRDA_TCTL, C-26
ORACLE_DRDA_TRACE, C-27
TRACE_LEVEL, C-27
LOG_DESTINATION, C-27
privileges
BIND
DB2/OS390, 13-1
DB2/UIDB, 13-4
BINDADD
DB2/OS390, 13-2
DB2/UIDB, 13-4
BINDAGENT
DB2/OS390, 13-2
CONNECT
DB2/UIDB, 13-4
COPY
DB2/OS390, 13-1
CREATE IN
DB2/OS390, 13-2
CREATE PUBLIC DATABASE LINK, 14-7
CREATEIN
DB2/UIDB, 13-4
CREATETAB
DB2/OS390, 13-2
DB2/UIDB, 13-4
EXECUTE
DB2/OS390, 13-1
DB2/UIDB, 13-4

R

RAW datatype
caution with DRDA_DEFAULT_CCSID, C-18
read-only gateway option
set with DRDA_READ_ONLY, C-24
rebind required with any change to
DRDA_DISABLE_CALL, C-19
DRDA_ISOLATION_LEVEL, C-21
DRDA_PACKAGE_COLLID, C-22
DRDA_PACKAGE_CONSTOKEN, C-23
DRDA_PACKAGE_NAME, C-23
DRDA_PACKAGE_OWNER, C-23
DRDA_PACKAGE_SECTIONS, C-24
recovery user ID and password
DB2/400, 13-3
DB2/OS390, 13-2
DB2/UIDB, 13-4
RECOVERY_ACCOUNT

- account username, 3-8, 5-8, 7-8, 9-8
- creating a recovery account, 3-8, 5-8, 7-8, 9-8
- remote
 - database
 - DB2/400, 13-3
 - DB2/OS390, 13-2
 - DB2/UDB, 13-4
 - DRDA_PACKAGE_SECTIONS, C-23
 - privileges of user/ ID, D-2
 - DRDA database, DRDA_ISOLATION_
 - LEVEL, C-21
 - transaction program
 - hardware memory requirements, 12-2
- remote functions
 - referenced in SQL statements, C-7
- requirements
 - hardware, 12-1
 - software, 12-3

S

- sample
 - SQL scripts, 14-7
- schema privileges - CREATEIN, 13-4
- security
 - overview, 15-1
- service port number, DRDA_CONNECT_
 - PARM, C-18
- shift attribute, multi-byte support, E-9
- SID, 3-1, 5-1, 7-1, 9-1, 11-1
- silent mode installation, A-4, A-5
- software requirements, 12-3
- SPUFI, a database native tool, 14-1
- SQL
 - statements, DRDA_ISOLATION_LEVEL, C-21
- stability, of cursor, DRDA_ISOLATION_
 - LEVEL, C-21
- Startup Shell Script
 - parameter, tailoring to configure the host, 14-2
- statement
 - CREATE DATABASE, client/server
 - configuration, E-3
- stored procedure
 - usage, C-19
- stored procedures
 - GTW\$_BIND_PKG, 14-7
- system privileges - BINDADD and BINDAGENT
 - DB2/OS390, 13-2

T

- TCP/IP
 - concurrent connections, 12-2
 - DRDA_CONNECT_PARM, C-18
- tnsnames.ora file, 17-2
- token, package consistency, DRDA_PACKAGE_
 - CONSTOKEN, C-22
- trace control, C-27
- TRACE_LEVEL parameter
 - defined, C-27

- tracing
 - code, C-27
 - LOG_DESTINATION, C-26
 - ORACLE_DRDA_TRACE, C-27
- Transaction log table
 - creating, 3-9, 5-9, 7-9, 9-9
- transaction mode, read-only, DRDA_READ_
 - ONLY, C-24
- transform, character set transforms with multi-byte
 - support, E-8
- transform, not required for DRDA Server, E-10
- troubleshooting, B-1
 - fatal errors, B-3
- Two-phase commit
 - configuration, 3-8, 5-8, 7-8, 9-8
 - transaction log table, 3-9, 5-9, 7-9, 9-9
- two-phase commit
 - ORACLE2PC table
 - DB2/400, 13-3
 - DB2/UDB, 13-4
 - ORACLE2PC table, DB2/OS390, 13-1

U

- user ID mapping
 - DB2/400, 15-3
 - DB2/OS390, 15-3

X

- X windows
 - display errors, B-1