

Oracle® Database Gateway
Installation and Configuration Guide
12c Release 1 (12.1) for Microsoft Windows
E17926-05

July 2014

Oracle Database Gateway Installation and Configuration Guide, 12c Release 1 (12.1) for Microsoft Windows
E17926-05

Copyright © 2006, 2014, Oracle and/or its affiliates. All rights reserved.

Primary Author: Maitreyee Chaliha

Contributing Author: Vira Goorah, Govind Lakkoju, Peter Wong, Juan Pablo Ahues-Vasquez, Peter Castro and Charles Benet

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

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

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

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

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

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

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

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

Contents

Preface	xi
Intended Audience.....	xi
Documentation Accessibility	xi
Related Documents	xi
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.....	1-2
Using Windows User Account as Oracle Home User	1-3
Oracle Database Gateway Upgrades	1-3
Accessing the Installation Software	1-3
Downloading Oracle Software from the OTN Web Site	1-3
Copying the Oracle Software	1-4
Running the Oracle Universal Installer	1-4
Installing and Configuring in Cluster Environments	1-4
Support for Single Client Access Name (SCAN).....	1-5
Local Listener.....	1-5
Load Balancing and Transparent Application Failover(TAF).....	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	2-2

Step Through the Oracle Universal Installer.....	2-3
--	-----

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	3-4
Configure the Oracle Database for Gateway Access.....	3-4
Configuring tnsnames.ora	3-4
Configuring tnsnames.ora for Multiple Listeners.....	3-5
Create Database Links.....	3-6
Configure Two-Phase Commit	3-6
Create a Recovery Account and Password.....	3-7
Create the Transaction Log Table	3-7
Create Sybase Views for Data Dictionary Support	3-8
Encrypt Gateway Initialization Parameter Values	3-8
Configure the Gateway to Access Multiple Sybase Databases.....	3-9
Multiple Sybase Databases Example: Configuring the Gateway.....	3-9
Multiple Sybase Databases Example: Configuring Oracle Net Listener.....	3-10
Multiple Sybase Databases Example: Stopping and Starting the Oracle Net Listener.....	3-10
Multiple Sybase Databases Example: Configuring Oracle Database for Gateway Access ..	3-10
Multiple Sybase Databases Example: Accessing Sybase Data.....	3-11

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	4-2
Step Through the Oracle Universal Installer.....	4-2

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	5-4
Configure the Oracle Database for Gateway Access.....	5-4
Configuring tnsnames.ora	5-4
Configuring tnsnames.ora for Multiple Listeners.....	5-5
Create Database Links.....	5-6
Configure Two-Phase Commit	5-6

Create a Recovery Account and Password.....	5-7
Create the Transaction Log Table.....	5-7
Encrypt Gateway Initialization Parameter Values	5-8
Configure the Gateway to Access Multiple Informix Databases	5-8
Multiple Informix Databases Example: Configuring the Gateway	5-9
Multiple Informix Databases Example: Configuring Oracle Net Listener	5-9
Multiple Informix Databases Example: Stopping and Starting the Oracle Net Listener	5-10
Multiple Informix Databases Example: Configuring Oracle Database for Gateway Access	5-10
Multiple Informix Databases Example: Accessing Informix Data.....	5-11

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	6-2
Step Through the Oracle Universal Installer.....	6-2

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	7-4
Configure the Oracle Database for Gateway Access.....	7-4
Configuring tnsnames.ora	7-5
Configuring tnsnames.ora for Multiple Listeners	7-5
Create Database Links.....	7-6
Configure Two-Phase Commit	7-6
Create a Recovery Account and Password.....	7-7
Create the Transaction Log Table	7-7
Encrypt Gateway Initialization Parameter Values	7-8
Configure the Gateway to Access Multiple Teradata Databases.....	7-8
Multiple Teradata Databases Example: Configuring the Gateway	7-9
Multiple Teradata Databases Example: Configuring Oracle Net Listener	7-9
Multiple Teradata Databases Example: Stopping and Starting the Oracle Net Listener	7-10
Multiple Teradata Databases Example: Configuring Oracle Database for Gateway Access	7-10
Multiple Teradata Databases Example: Accessing Teradata Data	7-11

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	8-2
	Step Through the Oracle Universal Installer	8-3
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	9-4
	Configure the Oracle Database for Gateway Access	9-4
	Configuring tnsnames.ora	9-5
	Configuring tnsnames.ora for Multiple Listeners	9-5
	Create Database Links	9-6
	Configure Two-Phase Commit	9-7
	Create a Recovery Account and Password	9-7
	Create the Transaction Log Table	9-8
	Create SQL Server Views for Data Dictionary Support	9-8
	Encrypt Gateway Initialization Parameter Values	9-9
	Configure the Gateway to Access Multiple SQL Server Databases	9-9
	Multiple SQL Server Databases Example: Configuring the Gateway	9-9
	Multiple SQL Server Databases Example: Configuring Oracle Net Listener	9-10
	Multiple SQL Server Databases Example: Stopping and Starting the Oracle Net Listener	9-10
	Multiple SQL Server Databases Example: Configuring Oracle Database for Gateway Access	9-10
	Multiple SQL Server Databases Example: Accessing SQL Server Data	9-11

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	10-2
	Step Through the Oracle Universal Installer	10-3
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	11-4
	Configure the Oracle Database for Gateway Access	11-5

Multiple DRDA Databases Example: Configuring Oracle Database for Gateway Access ..	14-9
Multiple DRDA Databases Example: Accessing DB2 Data	14-9

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
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 or Upgrade Data Dictionary Views.....	16-2

Part VIII Removing Oracle Database Gateway

17 Removing Oracle Database Gateway

About the Deinstallation Tool.....	17-1
Removing Oracle Software.....	17-2

Part IX Appendixes

A Using Response Files for Noninteractive Installation

Introduction.....	A-1
Using Response Files to Install Oracle Components in Noninteractive Mode	A-1
Customizing a Sample Response File	A-2
Creating a New Response File	A-2
Running Oracle Universal Installer and Specifying a Response File	A-3

B Oracle Database Gateway Troubleshooting

Verifying 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-2
Fatal Errors.....	B-2
Noninteractive Installation Response File Error Handling.....	B-3

Cleaning Up After a Failed Installation.....	B-3
--	-----

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-5
Oracle Database Gateway for ODBC Initialization Parameters	C-6
Oracle Database Gateway for DRDA Initialization Parameters.....	C-7
Initialization Parameter Description.....	C-8
HS_CALL_NAME	C-8
HS_DB_DOMAIN	C-8
HS_DB_INTERNAL_NAME	C-9
HS_DB_NAME	C-9
HS_DESCRIBE_CACHE_HWM	C-9
HS_LANGUAGE	C-10
HS_LONG_PIECE_TRANSFER_SIZE	C-11
HS_OPEN_CURSORS	C-11
HS_RPC_FETCH_REBLOCKING	C-11
HS_RPC_FETCH_SIZE	C-12
HS_FDS_SHAREABLE_NAME	C-12
HS_TIME_ZONE	C-12
HS_TRANSACTION_MODEL	C-13
IFILE	C-13
HS_FDS_CONNECT_INFO	C-14
HS_FDS_PROC_IS_FUNC.....	C-15
HS_FDS_RECOVERY_ACCOUNT	C-15
HS_FDS_RECOVERY_PWD.....	C-16
HS_FDS_RESULTSET_SUPPORT	C-16
HS_FDS_TRACE_LEVEL.....	C-16
HS_FDS_TRANSACTION_LOG	C-17
HS_FDS_REPORT_REAL_AS_DOUBLE	C-17
HS_FDS_FETCH_ROWS.....	C-17
HS_FDS_CAPABILITY.....	C-17
HS_FDS_ISOLATION_LEVEL.....	C-18
HS_FDS_PACKAGE_COLLID.....	C-18
HS_IDLE_TIMEOUT	C-19
HS_FDS_MBCS_TO_GRAPHIC	C-19
HS_FDS_GRAPHIC_TO_MBCS	C-19
HS_FDS_TIMESTAMP_MAPPING	C-20
HS_FDS_DATE_MAPPING	C-20
HS_FDS_ARRAY_EXEC	C-20
HS_FDS_QUOTE_IDENTIFIER.....	C-20
HS_NLS_LENGTH_SEMANTICS	C-21
HS_KEEP_REMOTE_COLUMN_SIZE.....	C-21
HS_FDS_REMOTE_DB_CHARSET	C-21

HS_FDS_SUPPORT_STATISTICS	C-22
HS_FDS_RSET_RETURN_ROWCOUNT	C-22
HS_FDS_SQLLEN_INTERPRETATION	C-22
HS_FDS_AUTHENTICATE_METHOD	C-23
HS_FDS_ENCRYPT_SESSION	C-23
HS_FDS_TRUSTSTORE_FILE.....	C-23
HS_FDS_TRUSTSTORE_PASSWORD.....	C-24

D Configuration Worksheet for DRDA

Index

Preface

This guide describes how to install Oracle Database Gateway for Sybase, Informix, Teradata, SQL Server, ODBC, and DRDA on Microsoft Windows (64-bit) platform.

This preface covers the following topics:

- [Intended Audience](#)
- [Documentation Accessibility](#)
- [Related Documents](#)
- [Conventions](#)

Intended Audience

This manual is intended for Oracle database administrators who perform the following tasks:

- Installing Oracle Database Gateways
- Configuring Oracle Database Gateways

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

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 SQL Server User's Guide*
- *Oracle Database Gateway for ODBC 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 Development 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 User's Guide*
- *Oracle Database Security Guide*

Conventions

The following text 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, user names, 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 the installation of Oracle Database Gateways on Microsoft Windows (64-bit), as well as issues that you should consider before installing the software. It includes the following topics:

- [Gateway Installation Configurations](#)
- [Gateway Installation Methods](#)
- [Installation Considerations](#)
- [Oracle Database Gateway Upgrades](#)
- [Accessing the Installation Software](#)
- [Running the Oracle Universal Installer](#)
- [Installing and Configuring in Cluster Environments](#)

Gateway Installation Configurations

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

1. On the same computer as an existing Oracle database but in a different Oracle home.
2. On a system with no Oracle database.
3. 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 Gateway, 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.

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](#)
- [Using Windows User Account as Oracle Home User](#)

Release Notes

Read the release notes for the product before installing it. The release notes are available on the Oracle Database 12c Release 1 (12.1) installation media. The latest version of the release notes is also available on the Oracle Technology Network (OTN) Web site:

<http://docs.oracle.com/>

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 My Oracle Support Web site for the most up-to-date list of certified hardware platforms and operating system versions. The My Oracle Support Web site is available at the following Web site:

<https://support.oracle.com>

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 in 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 12c Release 1 (12.1) software into an existing Oracle 10gR2 Oracle home directory. If you attempt to install this release in an Oracle home directory that contains software from an earlier Oracle release, then the installation will fail.

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

Using Windows User Account as Oracle Home User

With Windows, you log in to a user with Administrator privileges to install the Oracle Database software. You can also specify an Oracle Home User (based on a low-privileged, non-administrative user account) during installation.

The following are the Windows User Accounts:

- Windows Local User account
- Windows Domain User account
- Windows Managed Services Account (MSA)
- Windows Built-in Account

See Also: "Using Oracle Home User on Windows" in *Oracle Database Platform Guide for Microsoft Windows*

Oracle Database Gateway 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:

- [Downloading Oracle Software from the OTN Web Site](#)
- [Copying the Oracle Software](#)

Downloading Oracle Software from the OTN Web Site

You can download the installation files from the OTN and extract them to a local directory on your system.

To download the installation files:

1. Use any browser to access the OTN software download page:
<http://www.oracle.com/technetwork/indexes/downloads/index.html>
2. Navigate to each of the download pages for the product that you want to install.
3. On each download page, identify the required disk space by adding the file sizes for each required file. The file sizes are listed next to the file names.
4. Select a file system with enough free space to store and expand the files. In most cases, the available disk space must be at least twice the size of each compressed file.
5. On the file system that you just selected, create a parent directory for each product that you plan to install, for example Dg_1, to hold the installation directories.
6. Download all the installation files to the directories that you just created.
7. Verify that the files that you downloaded are the same size as the corresponding files on OTN.
8. Extract the files in each directory that you just created.
9. After you have extracted the required installation files, go to the "[Running the Oracle Universal Installer](#)" section on page 1-4.

Copying the Oracle Software

Before installing Oracle Database Gateway, you might want to copy the software to a local directory. This enables the installation process to run faster.

To copy the contents of the installation media to a local directory:

1. Create a directory on your hard drive. For example:
d:\install\Disk1
2. Copy the contents of the installation media to the directory that you just created.
3. After you have copied all the required installation files, go to the ["Running the Oracle Universal Installer"](#) section on page 1-4.

Running the Oracle Universal Installer

In most cases, you use the graphical user interface (GUI) provided by Oracle Universal Installer to install the gateway. However, you can also use Oracle Universal Installer to complete noninteractive installations, without using the GUI.

See Also: Refer to [Appendix A, "Using Response Files for Noninteractive Installation"](#) for information about noninteractive installations and other advanced installation topics

Start the Installer and install the software, as follows:

1. If you are installing from a local directory, then double-click `setup.exe` located in the directory you created for the downloaded or copied installation files.
2. When installing from the installation media, the Autorun screen automatically appears. If the Autorun screen does not appear, then:
 - a. From the **Start** menu, select **Run**.
 - b. Enter the following:

```
DRIVE_LETTER:\autorun\autorun.exe
```

In the Autorun screen, select **Install/Deinstall Products**.

3. Use the following guidelines to complete the installation:
 - Follow the instruction displayed in the Installer window. If you need additional information, click **Help**.
 - If you encounter errors while installing or linking the software, then see [Appendix B, "Oracle Database Gateway Troubleshooting"](#) for information about troubleshooting.
4. When the installation is complete, click **Exit**, then click **Yes** to exit from the Installer.

Installing and Configuring in Cluster Environments

Oracle Database Gateway can be installed in the existing Oracle Database home or in a separate gateway home, on all nodes. Oracle OUI can install Oracle Database Gateway on either all nodes or selective nodes.

Oracle recommends not to use the listener from the Oracle Database Gateway home. Instead configure the listener in Grid home. By default a local listener is created during

cluster configuration that runs out of the grid infrastructure home and listens on the specified port (default is 1521) of the node Virtual IP(VIP).

Support for Single Client Access Name (SCAN)

Oracle Database 11g Release 2 and higher clients connect to the database using Support for Single Client Access Name (SCAN). It provides a single name to the clients connecting to Oracle RAC that does not change throughout the life of the cluster, even if you add or remove nodes from the cluster. Clients connecting with SCAN can use a simple connection string, such as a thin JDBC URL or EZConnect, and achieve load balancing and client connection failover.

In addition to the three SCAN listeners (one per virtual IP address), there is a node listener on every node hosting a database instance. The purpose of using two layers of listeners (SCAN listeners and node listeners) is to separate the two functions of listeners in an Oracle RAC, firstly to load balance connections and secondly to spawn-and-bequeath sessions. The SCAN listeners will receive connection requests from clients, randomly distributed by the GNS (Grid Naming Services). The SCAN listener will then use load balancing metrics to redirect the request intelligently to the node listener on the node best able to offer the requested service. Database instances register with the SCAN listeners as remote listeners, and with the node listeners as local listeners.

Oracle Database Gateway can not be configured with SCAN, a single name for Database to connect to the gateway. There are two reasons for this. Gateway does not work with remote listeners. Unlike Database where you can specify `REMOTE_LISTENER` to set to the SCAN listener, there is no support for it in Oracle Database Gateways. This is essential for SCAN listener to route the connection to the node listener. Secondly, the gateway does not register with the cluster for it to be managed as a cluster resource.

For gateway, SCAN is not very useful when the Oracle Database and Oracle Database Gateway are running on the same cluster. Oracle Database Gateway can be installed and configured on each node where database is installed, and database can be configured such that each instance connect to the Gateway running on the same node.

Local Listener

Oracle Database Gateway service should be configured using the local listener. It is the local listener that spawns the gateway process. That means listener should know which gateway process to spawn. Use the `listener.ora` in Grid infrastructure home to add the Gateway SID. If a SCAN listener for Database is already running on that node, you can use the same `listener.ora` file to configure the local listener.

Load Balancing and Transparent Application Failover(TAF)

Oracle Database Gateway itself does not support either the client-side (using `tnsnames` in database home) or server-side (using SCAN) load balancing. Load balancing at the Gateway level is not applicable because the Gateway process is currently dedicated to a single session. However, by associating a different Oracle Database Gateway instance for each database instance, you can achieve node level load balancing, that is, selecting a least loaded node happens through load balancing on the database.

Oracle Database Gateway supports connection failover feature. If you configure client-side connection load balancing for Gateway, it works similar to failover.

Whenever database fail over happens, that is, session migrates from one database instance to other database instance (on a new node), the migrated session will use the gateway instance from that new node.

Three types of Oracle Net failover functionality are available by default to Oracle Call Interface (OCI) applications:

- **session:** Set to failover the session. If a user connection is lost, then a new session is automatically created for the user on the backup. This type of failover does not attempt to recover select operations.
- **select:** Set to enable users with open cursors to continue fetching on them after failure. However, this mode involves overhead on the client side in normal select operations.
- **none:** This is the default. No failover functionality is used. This can also be explicitly specified to prevent failover from happening.

For failover to work, `tnsnames.ora` in Database home need to be configured with multiple listener addresses.

If the instance fails after the connection, then the TAF application fails over to the other node's listener, reserving any `SELECT` statements in progress.

In the following example of `tnsnames.ora` for load balancing that only works as failover, the database connects to the gateway on host `gateway2-server` only if the gateway on `gateway1-server` is not available:

```
dg4sybs.us.example.com=
  (DESCRIPTION=
    (LOAD_BALANCE=on)
    (FAILOVER=on)
    (ADDRESS=
      (PROTOCOL=tcp)
      (HOST=gateway1-server)
      (PORT=1521))
    (ADDRESS=
      (PROTOCOL=tcp)
      (HOST=gateway2-server)
      (PORT=1521))
    (CONNECT_DATA=
      (SERVICE_NAME=dg4sybs.us.example.com) (HS=OK)
      (FAILOVER_MODE=
        (TYPE=select)
        (METHOD=basic))))
```

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.

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 2-3 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 2-2](#) 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 2-2

Hardware Requirements

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

Table 2-1 Hardware Requirements for Oracle Database Gateway for Sybase

Requirement	For Microsoft Windows (64-bit)
Total disk space	5 GB
Physical Memory (RAM)	Minimum of 1 GB

Table 2–1 (Cont.) Hardware Requirements for Oracle Database Gateway for Sybase

Requirement	For Microsoft Windows (64-bit)
Virtual memory	Double the amount of RAM
Video adapter	256 colors
Processor	AMD64, or Intel Extended memory (EM64T)

Checking the Hardware Requirements

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

1. Determine the physical RAM size. For a computer using Microsoft Windows 2000, for example, open **System** in the control panel and select the **General** tab. If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.
2. Determine the size of the configured swap space (also known as paging file size). For a computer using Microsoft Windows 2000, for example, open **System** in the control panel, select the **Advanced** tab, and click **Performance Options**.

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

3. Determine the amount of free disk space on the system. For a computer using Microsoft Windows 2000, for example, open **My Computer**, right-click the drive where the Oracle software is to be installed, and select **Properties**.
4. Determine the amount of disk space available in the `temp` directory. This is equivalent to the total amount of free disk space, minus what will be needed for the Oracle software to be installed.

If there is less than 125 MB of disk space available in the `temp` directory, then first delete all unnecessary files. If the temp disk space is still less than 125 MB, then set the `TEMP` or `TMP` environment variable to point to a different hard drive. For a computer using Microsoft Windows 2000, for example, open the **System** control panel, select the **Advanced** tab, and click **Environment Variables**.

Software Requirements

Oracle Database Gateway for Sybase is supported on the following Microsoft Windows (64-bit) operating systems:

- Microsoft Windows Server 2003 - all x64 editions
- Microsoft Windows Server 2003 R2 - all x64 editions
- Microsoft Windows XP Professional x64 Edition
- Microsoft Windows Vista x64 - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008 x64

Certified Configurations

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

<http://www.oracle.com/technetwork/database/gateways/index.html>

Step Through the Oracle Universal Installer

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

Table 2–2 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 <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, the Oracle Universal Installer displays the Available Products screen.
Oracle Universal Installer: Available Product Components	a. Select Oracle Database Gateway for Sybase 12.1 b. Click Next.
Oracle Database Gateway for Sybase	Sybase Database Server Host Name - Specify the host name of the machine hosting the Sybase database server. Sybase Database Server Port number - Specify the port number of the Sybase database server Sybase Database Name - Specify the Sybase database name Click Next to continue.
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.

The gateway is now installed.

When the Oracle Universal Installer confirms that the installation is complete, verify that the installation procedure was successful. To do this, read the contents of the installation 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

Each of these variables in the log file name represents the date and time the product was installed.

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 User'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
<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 that will be configured in the next step.

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

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>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:

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

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. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. If the service is already running, click **Stop** to stop it.
4. Click **Start** to start or restart the service.

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.

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:

Table 3–1 Gateway Parameters for tnsnames.ora File

Variable	Description
<i>connect_descriptor</i>	is the description of the object to connect to as specified when creating the database link, such as <i>dg4sybs</i> . Check the <i>sqlnet.ora</i> file for the following parameter setting: <code>names.directory_path = (TNSNAMES)</code> Note: The <i>sqlnet.ora</i> file is typically stored in <i>ORACLE_HOME\network\admin</i> .
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 <i>listener.ora</i> 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 <i>listener.ora</i> 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.

Configuring tnsnames.ora for Multiple Listeners

To ensure higher availability, you can specify multiple listeners within the connect descriptor.

```

connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_1)
      (PORT=port_number_1)
    )
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_2)
      (PORT=port_number_2)
    )
  )
  (CONNECT_DATA=
    (SID=gateway_sid)
  (HS=OK) )

```

This indicates that, if the listener for *host_name_1* and *port_number_1* is not available, then the second listener for *host_name_2* and *port_number_2* will take over.

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
<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 Sybase database, as follows:

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

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity User'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 User'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 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 User'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 3-7. 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 at the MS-DOS prompt, 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 that contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity User'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 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
> copy initdg4sybs.ora initdg4sybs2.ora
> copy 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.

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

Perform the following steps:

1. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. Click **Stop**.
4. Click **Start**.

Multiple Sybase Databases 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.

This example describes how to configure Oracle Net on the Oracle database for multiple gateway instances. It 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.

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](#)" section on page 4-2 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 4-2](#) 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 4-2

Hardware Requirements

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

Table 4-1 *Hardware Requirements for Oracle Database Gateway for Informix*

Requirement	For Microsoft Windows (64-bit)
Total disk space	5 GB
Physical Memory	Minimum of 1 GB
Virtual memory	Double the amount of RAM
Video adapter	256 colors

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

Requirement	For Microsoft Windows (64-bit)
Processor	AMD64, or Intel Extended memory (EM64T)

Checking the Hardware Requirements

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

1. Determine the physical RAM size. For a computer using Microsoft Windows 2000, for example, open **System** in the control panel and select the **General** tab. If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.
2. Determine the size of the configured swap space (also known as paging file size). For a computer using Microsoft Windows 2000, for example, open **System** in the control panel, select the **Advanced** tab, and click **Performance Options**.

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

3. Determine the amount of free disk space on the system. For a computer using Microsoft Windows 2000, for example, open **My Computer**, right-click the drive where the Oracle software is to be installed, and select **Properties**.
4. Determine the amount of disk space available in the `temp` directory. This is equivalent to the total amount of free disk space, minus what will be needed for the Oracle software to be installed.

If there is less than 125 MB of disk space available in the `temp` directory, then first delete all unnecessary files. If the temp disk space is still less than 125 MB, then set the `TEMP` or `TMP` environment variable to point to a different hard drive. For a computer using Microsoft Windows 2000, for example, open the **System** control panel, select the **Advanced** tab, and click **Environment Variables**.

Software Requirements

Oracle Database Gateway for Informix is supported on the following Microsoft Windows (64-bit) operating systems:

- Microsoft Windows Server 2003 - all x64 editions
- Microsoft Windows Server 2003 R2 - all x64 editions
- Microsoft Windows XP Professional x64 Edition
- Microsoft Windows Vista x64 - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008 x64

Certified Configurations

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

<http://www.oracle.com/technetwork/database/gateways/index.html>

Step Through the Oracle Universal Installer

Table 4–2 describes the installation procedure for Oracle Database Gateway for Informix.

Table 4–2 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 <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, the Oracle Universal Installer displays the Available Products screen.
Oracle Universal Installer: Available Product Components	a. Select Oracle Database Gateway for Informix 12.1. b. Click Next.
Oracle Database Gateway for Informix	Informix Database Server Host Name - Specify the host name of the machine hosting the Informix database server. Informix Database Server Port number - Specify the port number of the Informix database server Informix Server Name - Specify the Informix server name Informix Database Name - Specify the Informix database name Click Next to continue.
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.

The gateway is now installed.

When the Oracle Universal Installer confirms that the installation is complete, verify that the installation procedure was successful. To do this, read the contents of the installation 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

Each of these variables in the log file name represents the date and time the product was installed.

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 `dg4ifmx`.

You can define a gateway SID, but using the default of `dg4ifmx` 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>	specifies 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 User'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, 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 that will be configured in the next step.

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

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:

```

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

```

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. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. If the service is already running, click **Stop** to stop it.
4. Click **Start** to start or restart the service.

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.

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:

Table 5–1 Parameters for listener.ora File

Variable	Description
<i>connect_descriptor</i>	is the description of the object to connect to as specified when creating the database link, such as dg4ifmx. 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 <i>ORACLE_HOME</i> \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 5-3.
<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 5-2 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

Configuring tnsnames.ora for Multiple Listeners

To ensure higher availability, you can specify multiple listeners within the connect descriptor.

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_1)
      (PORT=port_number_1)
    )
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_2)
      (PORT=port_number_2)
    )
  )
  (CONNECT_DATA=
    (SID=gateway_sid)
  )
  (HS=OK))
```

This indicates that, if the listener for *host_name_1* and *port_number_1* is not available, then the second listener for *host_name_2* and *port_number_2* will take over.

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
<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 Informix database, as follows:

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

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity User'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 User'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 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 User'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 5-7. 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.

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 that contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity User'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
> copy initdg4ifmx.ora initdg4ifmx2.ora
> copy 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.

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

Perform the following steps:

1. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. Click **Stop**.
4. Click **Start**.

Multiple Informix Databases 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.

This example describes how to configure Oracle Net on the Oracle database for multiple gateway instances. It 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 of Oracle Database Gateway for Teradata.

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](#)" section on page 6-1.

2. Run the Oracle Universal Installer

See "[Step Through the Oracle Universal Installer](#)" section on page 6-2 for more information on 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 6-2](#) for a description of the installation procedure of Oracle Database Gateway for Teradata.

System Requirements for Oracle Database Gateway for Teradata

This chapter 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 6-2

Hardware Requirements

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

Table 6-1 *Hardware Requirements for Oracle Database Gateway for Teradata*

Requirement	For Microsoft Windows (64-bit)
Total disk space	5 GB
Physical Memory	Minimum of 1 GB
Virtual memory	Double the amount of RAM
Video adapter	256 colors

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

Requirement	For Microsoft Windows (64-bit)
Processor	AMD64, or Intel Extended memory (EM64T)

Checking the Hardware Requirements

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

1. Determine the physical RAM size. For a computer using Microsoft Windows 2000, for example, open **System** in the control panel and select the **General** tab. If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.
2. Determine the size of the configured swap space (also known as paging file size). For a computer using Microsoft Windows 2000, for example, open **System** in the control panel, select the **Advanced** tab, and click **Performance Options**.

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

3. Determine the amount of free disk space on the system. For a computer using Microsoft Windows 2000, for example, open **My Computer**, right-click the drive where the Oracle software is to be installed, and select **Properties**.
4. Determine the amount of disk space available in the `temp` directory. This is equivalent to the total amount of free disk space, minus what will be needed for the Oracle software to be installed.

If there is less than 125 MB of disk space available in the `temp` directory, then first delete all unnecessary files. If the temp disk space is still less than 125 MB, then set the `TEMP` or `TMP` environment variable to point to a different hard drive. For a computer using Microsoft Windows 2000, for example, open the **System** control panel, select the **Advanced** tab, and click **Environment Variables**.

Software Requirements

Oracle Database Gateway for Teradata is supported on the following Microsoft Windows (64-bit) operating systems:

- Microsoft Windows Server 2003 - all x64 editions
- Microsoft Windows Server 2003 R2 - all x64 editions
- Microsoft Windows XP Professional x64 Edition
- Microsoft Windows Vista x64 - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008 x64

Certified Configurations

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/technetwork/database/gateways/index.html>

Step Through the Oracle Universal Installer

Table 6–2 describes the installation procedure for Oracle Database Gateway for Teradata.

Table 6–2 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 <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, the Oracle Universal Installer displays the Available Products screen.
Oracle Database Gateway for Teradata	Teradata Database Server Host IP or Alias - Specify either the host IP or alias name of the machine running the Teradata database server. Teradata Database Server Port number - Specify the port number of the Teradata database server Teradata Database Name - Specify the Teradata database name Click Next to continue.
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.

The gateway is now installed.

When the Oracle Universal Installer confirms that the installation is complete, verify that the installation procedure was successful. To do this, read the contents of the installation 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

Each of these variables in the log file name represents the date and time the product was installed.

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=ip_address:port_number[/database_name]`

Where:

Variable	Description
<code>ip_address</code>	is the IP address of the machine hosting the Teradata database.
<code>port_number</code>	is the port number of the Teradata database server.
<code>database_name</code>	is the Teradata database name. The <code>database_name</code> variable is optional

See Also: [Appendix C, "Initialization Parameters"](#) and the *Oracle Database Heterogeneous Connectivity User'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
<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 that will be configured in the next step.

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4tera)
      (ENVS=PATH=oracle_home_directory\bin;teradata_client_
directory\lib;Windows_system_paths)
    )
  )
```

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>dg4tera</i>	specifies the executable name of the Oracle Database Gateway for Teradata.
<i>teradata_client_directory</i>	specifies the directory where the Teradata client resides.

Variable	Description
<code>Windows_system_paths</code>	<p>specifies the Microsoft Windows system paths.</p> <p>For example, <code>C:\gtwyhome\tg11\bin;C:\Program Files\NCR\Teradata Client\cliv2;C:\Program Files\NCR\Common Files\Shared ICU Libraries for Teradata\lib;C:\WINDOWS\system32;C:\WINDOWS</code></p>

If you already have an existing Oracle Net Listener, then add 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=dg4tera)
    (ENVS=PATH=oracle_home_directory\bin;teradata_client_directory\lib;Windows_
system_paths)
  )
)
```

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. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. If the service is already running, click **Stop** to stop it.
4. Click **Start** to start or restart the service.

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.

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>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.
<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 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.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

Configuring `tnsnames.ora` for Multiple Listeners

To ensure higher availability, you can specify multiple listeners within the connect descriptor.

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_1)
      (PORT=port_number_1)
    )
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_2)
      (PORT=port_number_2)
    )
  )
```

```

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

```

This indicates that, if the listener for *host_name_1* and *port_number_1* is not available, then the second listener for *host_name_2* and *port_number_2* will take over.

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
<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 Teradata database, as follows:

```

SQL> SELECT * FROM DUAL@dblink;

```

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity User'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 User'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 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 User'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.

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 7-7. 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 that contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity User'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 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
> copy initdg4tera.ora initdg4tera2.ora
> copy 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=PATH=oracle_home_directory\bin;teradata_client_directory\lib;Windows_
system_paths)
    )
    (SID_DESC=
        (SID_NAME=dg4tera2)
        (ORACLE_HOME=oracle_home_directory)
        (PROGRAM=dg4tera)
        (ENVS=PATH=oracle_home_directory\bin;teradata_client_directory\lib;Windows_
system_paths)
    )
    (SID_DESC=
        (SID_NAME=dg4tera3)
        (ORACLE_HOME=oracle_home_directory)
        (PROGRAM=dg4tera)
        (ENVS=PATH=oracle_home_directory\bin;teradata_client_directory\lib;Windows_
system_paths)
    )
)

```

where, `oracle_home_directory` is the directory where the gateway resides, `teradata_client_directory` specifies the directory where the Teradata client resides, and `Windows_system_paths` specifies the Microsoft Windows system paths.

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

Perform the following steps:

1. From the **Start** menu, select **Settings**, **Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. Click **Stop**.
4. Click **Start**.

Multiple Teradata Databases 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.

This example describes how to configure Oracle Net on the Oracle database for multiple gateway instances. It 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

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 of Oracle Database Gateway for SQL Server

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 SQL Server.

To install the Oracle Database Gateway for SQL Server, 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](#)" section on page 8-1.
2. Run the Oracle Universal Installer.

See "[Step Through the Oracle Universal Installer](#)" section on page 8-3 for more information on 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 8-2](#) for 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 8-2

Hardware Requirements

[Table 8-1](#) lists the minimum hardware requirements for Oracle Database Gateway for SQL Server.

Table 8-1 *Hardware Requirements for Oracle Database Gateway for SQL Server*

Requirement	For Microsoft Windows (64-bit)
Total disk space	5 GB
Physical Memory	Minimum of 1 GB
Virtual memory	Double the amount of RAM
Video adapter	256 colors

Table 8–1 (Cont.) Hardware Requirements for Oracle Database Gateway for SQL Server

Requirement	For Microsoft Windows (64-bit)
Processor	AMD64, or Intel Extended memory (EM64T)

Checking the Hardware Requirements

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

1. Determine the physical RAM size. For a computer using Microsoft Windows 2000, for example, open **System** in the control panel and select the **General** tab. If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.
2. Determine the size of the configured swap space (also known as paging file size). For a computer using Microsoft Windows 2000, for example, open **System** in the control panel, select the **Advanced** tab, and click **Performance Options**.

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

3. Determine the amount of free disk space on the system. For a computer using Microsoft Windows 2000, for example, open **My Computer**, right-click the drive where the Oracle software is to be installed, and select **Properties**.
4. Determine the amount of disk space available in the `temp` directory. This is equivalent to the total amount of free disk space, minus what will be needed for the Oracle software to be installed.

If there is less than 125 MB of disk space available in the `temp` directory, then first delete all unnecessary files. If the temp disk space is still less than 125 MB, then set the `TEMP` or `TMP` environment variable to point to a different hard drive. For a computer using Microsoft Windows 2000, for example, open the **System** control panel, select the **Advanced** tab, and click **Environment Variables**.

Software Requirements

Oracle Database Gateway for SQL Server is supported on the following Microsoft Windows (64-bit) operating systems:

- Microsoft Windows Server 2003 - all x64 editions
- Microsoft Windows Server 2003 R2 - all x64 editions
- Microsoft Windows XP Professional x64 Edition
- Microsoft Windows Vista x64 - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008 x64

Certified Configurations

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

<http://www.oracle.com/technetwork/database/gateways/index.html>

Step Through the Oracle Universal Installer

Step Through the Oracle Universal Installer

Table 8–2 describes the installation procedure for Oracle Database Gateway for SQL Server

Table 8–2 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 <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, the Oracle Universal Installer displays the Available Products screen.
Oracle Database Gateway for SQL Server	SQL Server Database Server Host Name - Specify the host name of the machine hosting the SQL Server database. SQL Server Database Server Port number - Specify the port number of the SQL Server database server SQL Server Database Name - Specify the SQL Server database name Click Next to continue.
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.

The gateway is now installed.

When the Oracle Universal Installer confirms that the installation is complete, verify that the installation procedure was successful. To do this, read the contents of the installation 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

Each of these variables in the log file name represents the date and time the product was installed.

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/[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>instance_name</i>	is the instance of SQL Server running on the machine.
<i>database_name</i>	is the SQL Server Database database name.

Both *instance_name* and *database_name* are optional. If *instance_name* is omitted and *database_name* is provided, the slash (/) is required. This can be shown as follows:

```
HS_FDS_CONNECT_INFO= host_name//database_name
```

See Also: [Appendix C, "Initialization Parameters"](#) and *Oracle Database Heterogeneous Connectivity User'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.

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=dg4msql)
    )
  )
```

Where:

Variable	Description
<code>gateway_sid</code>	specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <code>tnsnames.ora</code> file.
<code>oracle_home_directory</code>	specifies the Oracle home directory where the gateway resides.
<code>dg4msql</code>	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:

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

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. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. If the service is already running, click **Stop** to stop it.
4. Click **Start** to start or restart the service.

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.

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>	<p>is the description of the object to connect to as specified when creating the database link, such as <code>dg4msql</code>.</p> <p>Check the <code>sqlnet.ora</code> file in the Oracle database's <code>ORACLE_HOME</code> for the following lines:</p> <ul style="list-style-type: none"> ■ <code>names.directory_path = (TNSNAMES, HOSTNAME)</code> ■ <code>names.default_domain = world</code> ■ <code>name.default_zone = world</code> <p>Note: If the Oracle database is on Microsoft Windows, the file is <code>ORACLE_HOME\network\admin\sqlnet.ora</code>.</p> <p>If the <code>sqlnet.ora</code> file has these lines, <code>connect_descriptor</code> must end with the extension <code>.world</code>.</p>
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 9-3.
<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 9-3 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

Configuring `tnsnames.ora` for Multiple Listeners

To ensure higher availability, you can specify multiple listeners within the connect descriptor.

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
```

```

        (HOST=host_name_1)
        (PORT=port_number_1)
    )
    (ADDRESS=
        (PROTOCOL=TCP)
        (HOST=host_name_2)
        (PORT=port_number_2)
    )
    (CONNECT_DATA=
        (SID=gateway_sid)
    (HS=OK) )

```

This indicates that, if the listener for *host_name_1* and *port_number_1* is not available, then the second listener for *host_name_2* and *port_number_2* will take over.

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.
<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 SQL Server database, as follows:

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

See Also: *Oracle Database Administrator's Guide* and *Oracle Database Heterogeneous Connectivity User'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 User'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 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 User'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 9-7. 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 at the MS-DOS prompt, 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 that contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity User'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 `dg4msql1`
- 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
> copy initdg4msql.ora initdg4msql2.ora
> copy 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//db2
```

For `initdg4msql3.ora`, enter the following:

HS_FDS_CONNECT_INFO=204.179.79.15//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.

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

Perform the following steps:

1. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. Click **Stop**.
4. Click **Start**.

Multiple SQL Server Databases 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.

This example describes how to configure Oracle Net on the Oracle database for multiple gateway instances. It 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 Microsoft Windows.

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 Oracle Database Gateway for ODBC, 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"](#) section on page 10-3 for more information about running the Oracle Universal Installer.

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

See [Table 10-2](#) 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 Oracle Database Gateway for ODBC. It contains the following sections:

- ["Hardware Requirements"](#) on page 10-1
- ["Software Requirements"](#) on page 10-2

Hardware Requirements

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

Table 10-1 *Hardware Requirements for Oracle Database Gateway for ODBC*

Requirement	For Microsoft Windows (64-bit)
Total disk space	5 GB
Physical Memory	Minimum of 1 GB
Virtual memory	Double the amount of RAM

Table 10–1 (Cont.) Hardware Requirements for Oracle Database Gateway for ODBC

Requirement	For Microsoft Windows (64-bit)
Video adapter	256 colors
Processor	AMD64, or Intel Extended memory (EM64T)

Checking the Hardware Requirements

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

1. Determine the physical RAM size. For a computer using Microsoft Windows 2000, for example, open **System** in the control panel and select the **General** tab. If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.
2. Determine the size of the configured swap space (also known as paging file size). For a computer using Microsoft Windows 2000, for example, open **System** in the control panel, select the **Advanced** tab, and click **Performance Options**.

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

3. Determine the amount of free disk space on the system. For a computer using Microsoft Windows 2000, for example, open **My Computer**, right-click the drive where the Oracle software is to be installed, and select **Properties**.
4. Determine the amount of disk space available in the `temp` directory. This is equivalent to the total amount of free disk space, minus what will be needed for the Oracle software to be installed.

If there is less than 125 MB of disk space available in the `temp` directory, then first delete all unnecessary files. If the `temp` disk space is still less than 125 MB, then set the `TEMP` or `TMP` environment variable to point to a different hard drive. For a computer using Microsoft Windows 2000, for example, open the **System** control panel, select the **Advanced** tab, and click **Environment Variables**.

Software Requirements

Oracle Database Gateway for ODBC is supported on the following Microsoft Windows (64-bit) operating systems:

- Microsoft Windows Server 2003 - all x64 editions
- Microsoft Windows Server 2003 R2 - all x64 editions
- Microsoft Windows XP Professional x64 Edition
- Microsoft Windows Vista x64 - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008 x64

Certified Configurations

For the latest certified configuration refer to the OTN Web site:

<http://www.oracle.com/technetwork/database/gateways/index.html>

Step Through the Oracle Universal Installer

Table 10–2 describes the installation procedure for Oracle Database Gateway for ODBC.

Table 10–2 The Oracle Universal Installer: Steps for Installing Oracle Database Gateway for ODBC

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 <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, the Oracle Universal Installer displays the Available Products screen.
Oracle Universal Installer: Available Product Components	a. Select Oracle Database Gateway for ODBC 12.1 . 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.

The gateway is now installed.

When the Oracle Universal Installer confirms that the installation is complete, verify that the installation procedure was successful. To do this, read the contents of the installation 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

Each of these variables in the log file name represents the date and time the product was installed.

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` initialization parameter. 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 name of the system DSN defined in the Microsoft Windows ODBC Data Source Administrator.

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

Assume that a system DSN has been defined in the Microsoft Windows ODBC Data Source Administrator. In order to connect to this SQL Server database through the gateway, the following line is required in the `initsid.ora` file:

```
HS_FDS_CONNECT_INFO=sqlserver7
```

sqlserver7 is the name of the system DSN defined in the Microsoft Windows ODBC Data Source Administrator.

The following procedure enables you to define a system DSN in the Microsoft Windows ODBC Data Source Administrator, version 3.5:

1. From the **Start** menu, choose **Settings > Control Panel** and select **ODBC**.
2. Select the **System DSN** tab page to display the system data sources.
3. Click **Add**.
4. From the list of installed ODBC drivers, select the name of the driver that the data source will use. For example, select **SQL Server**.
5. Click **Finish**.
6. Enter a name for the DSN and an optional description. Enter other information depending on the ODBC driver. For example, for SQL Server enter the SQL Server machine name.

Note: The name entered for the DSN must match the value of the initialization parameter `HS_FDS_CONNECT_INFO` that is specified in `initsid.ora`.

7. Refer to your ODBC driver documentation and follow the prompts to complete configuration of the DSN.

8. After creating the system DSN, click **OK** to exit the ODBC Data Source Administrator.

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.
<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 that will be configured in the next step.

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4odbc)
    )
  )
```

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 tnsnames.ora file.
<i>oracle_home_directory</i>	specifies the Oracle home directory where the gateway resides.
<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:

```
SID_LIST_LISTENER=
  (SID_LIST=
    (SID_DESC=.
      .
    )
    (SID_DESC=.
      .
    )
    (SID_DESC=
      (SID_NAME=gateway_sid)
      (ORACLE_HOME=oracle_home_directory)
      (PROGRAM=dg4odbc)
    )
  )
```

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. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. If the service is already running, click **Stop** to stop it.
4. Click **Start** to start or restart the service.

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:

Table 11-1 Gateway Parameters for tnsnames.ora File

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>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.
<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 11-3.
<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 11-3 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

Configuring tnsnames.ora for Multiple Listeners

To ensure higher availability, you can specify multiple listeners within the connect descriptor.

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_1)
      (PORT=port_number_1)
    )
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_2)
      (PORT=port_number_2)
    )
  )
  (CONNECT_DATA=
    (SID=gateway_sid)
  )
  (HS=OK)
```

This indicates that, if the listener for *host_name_1* and *port_number_1* is not available, then the second listener for *host_name_2* and *port_number_2* will take over.

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 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 User'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 that contain sensitive values can be stored in an encrypted form.

See Also: *Oracle Database Heterogeneous Connectivity User'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 names of the system DSN defined in the Microsoft Windows ODBC Data Source Administrator, 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
> copy initdg4odbc.ora initdg4odbc2.ora
> copy 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:

```
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4odbc)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4odbc)
  )
  (SID_DESC=
    (SID_NAME=dg4odbc2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4odbc)
  )
  (SID_DESC=
    (SID_NAME=dg4odbc3)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4odbc)
  )
)
```

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

Multiple ODBC Data Sources Example: Stopping and Starting the Oracle Net Listener

Perform the following steps:

1. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. Click **Stop**.
4. Click **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 of Oracle Database Gateway for DRDA.

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 guides you through 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. Log on to your host computer as a member of the Administrators group.
3. Run the Oracle Universal Installer.

See "[Step Through the Oracle Universal Installer](#)" section on page 12-2 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 12-2](#) 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 12-2

Hardware Requirements

[Table 12-1](#) lists the minimum hardware requirements for Oracle Database Gateway for DRDA.

Table 12-1 *Hardware Requirements for Oracle Database Gateway for DRDA*

Requirement	For Microsoft Windows (64-bit)
Total disk space	5 GB
Physical Memory	Minimum of 1 GB
Virtual memory	Double the amount of RAM
Video adapter	256 colors

Table 12–1 (Cont.) Hardware Requirements for Oracle Database Gateway for DRDA

Requirement	For Microsoft Windows (64-bit)
Processor	AMD64, or Intel Extended memory (EM64T)

Checking the Hardware Requirements

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

1. Determine the physical RAM size. For a computer using Microsoft Windows 2000, for example, open **System** in the control panel and select the **General** tab. If the size of the physical RAM installed in the system is less than the required size, then you must install more memory before continuing.
2. Determine the size of the configured swap space (also known as paging file size). For a computer using Microsoft Windows 2000, for example, open **System** in the control panel, select the **Advanced** tab, and click **Performance Options**.

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

3. Determine the amount of free disk space on the system. For a computer using Microsoft Windows 2000, for example, open **My Computer**, right-click the drive where the Oracle software is to be installed, and select **Properties**.
4. Determine the amount of disk space available in the `temp` directory. This is equivalent to the total amount of free disk space, minus what will be needed for the Oracle software to be installed.

If there is less than 125 MB of disk space available in the `temp` directory, then first delete all unnecessary files. If the temp disk space is still less than 125 MB, then set the `TEMP` or `TMP` environment variable to point to a different hard drive. For a computer using Microsoft Windows 2000, for example, open the **System** control panel, select the **Advanced** tab, and click **Environment Variables**.

Software Requirements

Oracle Database Gateway for DRDA is supported on the following Microsoft Windows (64-bit) operating systems:

- Microsoft Windows Server 2003 - all x64 editions
- Microsoft Windows Server 2003 R2 - all x64 editions
- Microsoft Windows XP Professional x64 Edition
- Microsoft Windows Vista x64 - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008 x64

Certified Configurations

Oracle continually updates supported gateway configurations. For the latest supported configuration information, visit the OTN Web site:

<http://www.oracle.com/technetwork/database/gateways/index.html>

Step Through the Oracle Universal Installer

[Table 12–2](#) describes the installation procedure for Oracle Database Gateway for DRDA.

Table 12–2 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 <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, the Oracle Universal Installer displays the Available Products screen.
Oracle Universal Installer: Available Product Components	a. Select Oracle Database Gateway for DRDA 12.1. 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.

The gateway is now installed.

When the Oracle Universal Installer confirms that the installation is complete, verify that the installation procedure was successful. To do this, read the contents of the installation 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

Each of these variables in the log file name represents the date and time the product was installed.

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 UDB for z/OS
- DB2 UDB for iSeries
- DB2 UDB for Linux, UNIX, and Windows

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 UDB for z/OS"](#) on page 13-1
- ["Configuring the DRDA Server for DB2 UDB for iSeries"](#) on page 13-3
- ["Configuring the DRDA Server for DB2 UDB for Linux, UNIX, and Windows"](#) on page 13-3
- ["Manual Binding of DRDA Gateway Packages"](#) on page 13-4

Configuring the DRDA Server for DB2 UDB for z/OS

Perform the following tasks to configure the DRDA server with DB2 on a z/OS system:

1. Define the user ID that owns the package

During first gateway usage for a particular DRDA server, an Oracle supplied package will be automatically bound to the DRDA server. The user ID and password that are used (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. The followings are minimum authorities needed by this user:

- Package privileges of `BIND`, `COPY`, and `EXECUTE`, for example:

```
GRANT BIND    ON PACKAGE oraclegtw.* TO userid
```

```
GRANT COPY      ON PACKAGE oraclegtw.* TO userid
GRANT EXECUTE  ON PACKAGE oraclegtw.* TO PUBLIC
```

- Collection privilege of CREATE IN, for example:

```
GRANT CREATE IN COLLECTION oraclegtw 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 HS_TRANSACTION_LOG 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 HS_FDS_RECOVERY_ACCOUNT and HS_FDS_RECOVERY_PWD parameters. If a distributed transaction fails, then the recovery process connects to the remote database using the user ID and password that are 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 HS_FDS_RECOVERY_ACCOUNT, then the gateway attempts to connect to a user ID of RECOVER when a distributed transaction is in doubt.

Determine the user ID and password that 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 that 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. Properly authorized users can do this with a utility such as 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 15, "Security Considerations"](#), for additional CDB tables and security information.

Configuring the DRDA Server for DB2 UDB for iSeries

Experience with DB2 UDB for iSeries 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 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 `HS_TRANSACTION_LOG` 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 `CRTSQLPKG` command
- Change authority on the library in which the package will be created

Choose a user ID now that will own the package and the `HS_TRANSACTION_LOG` table. Ensure that this user ID is defined in DB2 UDB for iSeries 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 `HS_FDS_RECOVERY_ACCOUNT` and `HS_FDS_RECOVERY_PWD` parameters. If a distributed transaction fails, then the recovery process connects to the remote database using the user ID and password that are 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 `HS_FDS_RECOVERY_ACCOUNT`, then the gateway attempts to connect to a user ID of `RECOVER` when a distributed transaction is in doubt.

Determine the user ID and password that you will use for recovery.

3. Determine DRDA location name for DB2 UDB for iSeries 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 for Linux, UNIX, and Windows

Experience with DB2/UDB, configuring the communication subsystem of DB2 UDB for Linux, UNIX, and Windows, and the host System Administration tools is required to perform the following steps.

1. Define the user ID that owns the package

During first gateway usage for a particular DRDA server, an Oracle supplied package will be automatically bound to the DRDA server. The user ID and password that are used (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. The followings are minimum authorities needed by this user:

- Package privileges of BIND and EXECUTE, for example:

```
GRANT BIND    ON PACKAGE oraclegtw.* TO userid
GRANT EXECUTE ON PACKAGE oraclegtw.* TO PUBLIC
```

- Schema privilege of CREATEIN, for example:

```
GRANT CREATEIN ON SCHEMA otgdb2 TO USER userid
GRANT CREATEIN ON SCHEMA oraclegtw 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 that will own the package and HS_TRANSACTION_LOG 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 HS_FDS_RECOVERY_ACCOUNT and HS_FDS_RECOVERY_PWD parameters. If a distributed transaction fails, then the recovery process connects to the remote database using the user ID and password that are 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 HS_FDS_RECOVERY_ACCOUNT, then the gateway attempts to connect to a user ID of RECOVER when a distributed transaction is in doubt.

Determine the user ID and password that you will use for recovery.

3. Determine DRDA location name for DB2 UDB for Linux, UNIX, and Windows 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.

Manual Binding of DRDA Gateway Packages

The gateway uses several DB2 packages, which it normally uploads and binds during the first time the gateway connects to a DB2 instance. In some customer environments, the connecting userid may not have the necessary privileges to perform the binding, or some customers may prefer to manually bind the packages rather than allow the gateway to do the binding.

In such cases, Oracle provides a predefined set of packages for manual binding. These packages come with several restrictions that must be observed by setting specific

gateway initialization parameters to set values otherwise, the gateway will attempt to rebind the package automatically.

This section contains the following sub-sections:

- [Manually Binding of Packages for DB2 UDB for z/OS](#)
- [Manually Binding of Packages for DB2 UDB for Linux, Unix, and Windows](#)

Manually Binding of Packages for DB2 UDB for z/OS

Perform the following steps to manually bind packages for DB2 UDB for z/OS:

1. Allocate a sequential dataset on z/OS using the parameters DSORG=PS, RECFM=FB, LRECL=80, and BLKSIZE=3120. For example,

```
userid.DBRMFILE.XMIT
```

2. Allocate a Partitioned DataSet using the parameters DSORG=PO, RECFM=FB, LRECL=80, and BLKSIZE=6160. for example,

```
userid.TG4DRDA.CNTL
```

3. FTP the following file to the previously allocated sequential dataset in BINARY mode:

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_dbrm.xmit
```

Use the PUT command to replace the sequential dataset contents.

4. FTP the following file to the previously allocated PDS in ASCII mode:

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_bind.jcl
```

Use the PUT command to place the file into the PDS as member name BIND.

5. Use the TSO command option of ISPF (option 6) to issue the RECEIVE command:

```
RECEIVE INDS('userid.DBRMFILE.XMIT')
```

Specify DA(userid.DDODBC.DBRMLIB) as the parameters to the RECEIVE command. This will unpack the xmit file and create the specified PDS name.

6. Edit the BIND JCL (userid.TG4DRDA.CNTL(BIND)) and follow the instructions to update the JCL. Once updated, submit the JCL to perform the actual binding of the packages and granting of execution privileges on the packages.

To use these packages with the gateway, please set the following init parameters in the gateway initialization file:

- HS_OPEN_CURSORS=200
- HS_FDS_PACKAGE_COLLID=NULLLID

Manually Binding of Packages for DB2 UDB for Linux, Unix, and Windows

Perform the following steps to manually bind packages for DB2 UDB for Linux, Unix, and Windows:

1. Copy the following files to the host running the DB2 instance from the *ORACLE_HOME\dg4db2\admin* directory:

```
DDOC510A.bnd
DDOC510B.bnd
DDOC510C.bnd
```

```
DDON510A.bnd  
DDON510B.bnd  
DDON510C.bnd  
DDOR510A.bnd  
DDOR510B.bnd  
DDOR510C.bnd  
DDOS510A.bnd  
DDOS510B.bnd  
DDOS510C.bnd  
DDOU510A.bnd  
DDOU510B.bnd  
DDOU510C.bnd  
dg4db2_luw_pkglist.lst
```

If copying via FTP, then files ending in .bnd should be transferred in BINARY mode and files ending in .lst should be transferred in ASCII mode.

2. Connect to the DB2 instance and issue the bind command. For example,

```
db2 'connect to <database_name> user <userid> using <password>'  
db2 'bind @dg4db2_luw_pkglist.lst grant public'
```

To use these packages with the gateway, set the following initialization parameters in the gateway initialization file:

- HS_OPEN_CURSORS=200
- HS_FDS_PACKAGE_COLLID=NULLID

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. [Create Tables and Views for Data Dictionary Support](#)
5. [Configure the Oracle Database for Gateway Access](#)
6. [Create Database Links](#)
7. [Configure the Gateway to Access Multiple DRDA Databases](#)

SQL scripts are provided to perform steps such as creating the `HS_TRANSACTION_LOG` 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 UDB for Linux, UNIX, and Windows), 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.

SQL scripts are located in the `dg4db2/admin` directory. Appropriate platform scripts are designated by having the DB2 platform identifiers (eg: "zos", "as400" and "luw") and version specific numbers (eg: vw7, vw8) in their file names.

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

Tailor the parameter file with additional parameters as needed. Refer to [Appendix C, "Initialization Parameters"](#) for a list of supported initialization parameters. Also refer to [Chapter 15, "Security Considerations"](#) for security aspects to tailoring the 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\dg4db2\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 syntax of the address on which the Oracle Net Listener listens using the TCP/IP protocol adapter is as follows:

```
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.

Note: You must use the same SID value in the `listener.ora` file and as 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=dg4db2)
)
)
```

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>dg4db2</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=dg4db2)
)
)
```

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. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. If the service is already running, click **Stop** to stop it.
4. Click **Start** to start or restart the service.

Configure Two-Phase Commit

Support for Two-Phase Commit requires running the `ORACLE_HOME\dg4db2\admin\dg4db2_tx.sql` script on the DB2 server. This script will create objects used by the gateway for Two-Phase Commit. Edit the script and replace the default recover account schema ("RECOVER") with the account name specified for the `HS_FDS_RECOVERY_ACCOUNT` initialization parameter. Refer to [Appendix C, "Initialization Parameters"](#) for more details.).

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 for Linux, UNIX, and Windows, then run

```
ORACLE_HOME\dg4db2\admin\dg4db2_luw_drop.sql
```

- If connecting to DB2 UDB for z/OS, then run

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_drop.sql
```

- If connecting to DB2 UDB for iSeries, then run

```
ORACLE_HOME\dg4db2\admin\dg4db2_as400_drop.sql
```

2. Create the data dictionary tables

Run the appropriate script to create the data dictionary tables.

- If connecting to DB2 UDB for Linux, UNIX, and Windows, then run

```
ORACLE_HOME\dg4db2\admin\dg4db2_luw_tab.sql
```

- If connecting to DB2 UDB for z/OS, then run

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_tab.sql
```

- If connecting to DB2 UDB for iSeries, then run

```
ORACLE_HOME\dg4db2\admin\dg4db2_as400_tab.sql
```

3. Create the data dictionary views

Run the appropriate script to create the data dictionary views:

- If connecting to DB2 UDB for Linux, UNIX, and Windows, then run

For DB2 UDB for Linux, UNIX, and Windows V7:

```
ORACLE_HOME\dg4db2\admin\dg4db2_luw_vw7.sql
```

For DB2 UDB for Linux, UNIX, and Windows V8:

```
ORACLE_HOME\dg4db2\admin\dg4db2_luw_vw8.sql
```

- If connecting to DB2 UDB for z/OS then run

For DB2 UDB for z/OS V7 (RACF security):

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_vw7r.sql
```

For DB2 UDB for z/OS V7 (DB2 security):

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_vw7s.sql
```

For DB2 UDB for z/OS V8 (RACF security):

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_vw8r.sql
```

For DB2 UDB for z/OS V8 (DB2 security):

```
ORACLE_HOME\dg4db2\admin\dg4db2_zos_vw8s.sql
```

- If connecting to DB2 UDB for iSeries, then run

For DB2 UDB for iSeries V5.1:

```
ORACLE_HOME\dg4db2\admin\dg4db2_as400_vw51.sql
```

For DB2 UDB for iSeries V5.2:

```
ORACLE_HOME\dg4db2\admin\dg4db2_as400_vw52.sql
```

For DB2 UDB for iSeries V5.3:

```
ORACLE_HOME\dg4db2\admin\dg4db2_as400_vw53.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 `tnsnames.ora` entry (`tnsnames.ora.sample`) is available in the `ORACLE_HOME\dg4db2\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:

Table 14–1 Gateway Parameters for tnsnames.ora File

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>dg4db2</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.
<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 14-2.
<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 14-2 for more information.
(HS=OK)	specifies that this connect descriptor connects to a non-Oracle system.

Configuring tnsnames.ora for Multiple Listeners

To ensure higher availability, you can specify multiple listeners within the connect descriptor.

```
connect_descriptor=
  (DESCRIPTION=
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_1)
      (PORT=port_number_1)
    )
    (ADDRESS=
      (PROTOCOL=TCP)
      (HOST=host_name_2)
      (PORT=port_number_2)
    )
    (CONNECT_DATA=
      (SID=gateway_sid)
    )
  )
```

(HS=OK)

This indicates that, if the listener for *host_name_1* and *port_number_1* is not available, then the second listener for *host_name_2* and *port_number_2* will take over.

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 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 User'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.
- 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:

- dg4db22 for the gateway accessing database db2.
- dg4db23 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\dg4db2\admin\initdg4db2.ora`, twice, naming one with the gateway SID for db2 and the other with the gateway SID for db3:

```
> cd ORACLE_HOME\dg4db2\admin
> copy initdg4db2.ora initdg4db22.ora
> copy initdg4db2.ora initdg4db23.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=dg4db2)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4db2)
  )
  (SID_DESC=
    (SID_NAME=dg4db22)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4db2)
  )
  (SID_DESC=
    (SID_NAME=dg4db23)
    (ORACLE_HOME=oracle_home_directory)
    (PROGRAM=dg4db2)
  )
)
```


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

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

Perform the following steps:

1. From the **Start** menu, select **Settings, Control Panel** and then select **Services**.
2. Select the Oracle Net Listener service for the gateway.
3. Click **Stop**.
4. Click **Start**.

Multiple DRDA Databases 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.

This example describes how to configure Oracle Net on the Oracle database for multiple gateway instances. It 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=dg4db2) )
  (HS=OK) )
new_db2_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4db22) )
  (HS=OK) )
new_db3_using=(DESCRIPTION=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name) )
  (CONNECT_DATA=
    (SID=dg4db23) )
  (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 `dg4db22` 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 dg4db23 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. Each DRDA database server with GRANTS and related native authorization mechanisms based on user ID provides control over database object access.

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 server's 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 11g 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. Any user ID can use a public database link. Only the user who created it can use a private database link. 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 UDB for z/OS

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.

A user with update privilege can update the CDB tables 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 UDB for iSeries

DB2 UDB for iSeries does not provide a user ID mapping capability comparable to that in DB2 UDB for z/OS. Normally, the user ID in an incoming DRDA connection request must be a valid user ID on that DB2 UDB for iSeries.

The DB2 UDB for iSeries 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 UDB for iSeries, 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 UDB for Linux, UNIX, and Windows

DB2 UDB for Linux, UNIX, and Windows does not provide a user ID mapping capability comparable to that in DB2 UDB for z/OS. 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. Initialization parameters that contain sensitive values might be stored in an encrypted form with it. Refer to Section 4.2.3, 'Encrypting Initialization parameters' in the *Oracle Database Heterogeneous Connectivity User's Guide* for more information about how to use the feature.

See Also: The parameters `HS_FDS_RECOVERY_ACCOUNT` and `HS_FDS_RECOVERY_PWD` in [Appendix C, "Initialization Parameters"](#) 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. [Gateway Initialization Parameter File](#)
3. [Update the Initialization Parameters](#)
4. [Bind Gateway Package](#)
5. [Install or Upgrade Data Dictionary Views](#)

Install the New Release

Install the new release of the gateway in a separate directory, as discussed in [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.

Gateway Initialization Parameter File

This release of Database Gateway for DRDA has a completely new architecture. Most of the prior parameters are obsolete. You should not use the old initialization file as a base and try to modify it. Instead, you should use the new initialization generated as part of installation as a base. Refer to [Appendix C, "Initialization Parameters"](#) for the syntax of the parameters.

Existing TG4DB2 customer migrating to this release of Database Gateway for DRDA would need to provide the recovery user's password `HS_FDS_RECOVERY_PWD`, with this release of gateway.

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 parameter.

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 or Upgrade Data Dictionary Views

For the correct functioning of the gateway, the data dictionary views must be installed on any DB2 system that will be accessed by the gateway. If you are upgrading the gateway, then you must replace the data dictionary views to the ones shipped with the latest gateway. The new views are required for the correct functioning of the latest version of the gateway. They contain necessary backward functionality to be compatible with the previous versions. 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 Gateways.

It contains the following chapter:

- [Chapter 17, "Removing Oracle Database Gateway"](#)

Removing Oracle Database Gateway

This chapter describes how to remove Oracle Database Gateway from an Oracle home directory. It contains information about the following topics:

- [About the Deinstallation Tool](#)
- [Removing Oracle Software](#)

About the Deinstallation Tool

The Deinstallation Tool (`deinstall`) is available in the installation media before installation, and is available in Oracle home directories after installation. It is located in `ORACLE_HOME\deinstall`.

The `deinstall` command stops Oracle software, and removes Oracle software and configuration files on the operating system.

The script uses the following syntax, where variable content is indicated by italics:

```
deinstall -home complete path of Oracle home [-silent] [-checkonly] [-local]
[-paramfile complete path of input parameter property file] [-params name1=value
name2=value . . .] [-o complete path of directory for saving files] [-help | -h]
```

The options are:

- `-silent`

Use this flag to run the command in noninteractive mode. This option requires a properties file that contains the configuration values for the Oracle home that is being deinstalled or deconfigured.

To create a properties file and provide the required parameters, see the template file `deinstall.rsp.tmpl`, located in the response folder. If you prefer, instead of using the template file, you can generate a properties file by using the `-checkonly` option to have `deconfig` discover information from the Oracle home that you want to deinstall and deconfigure. The tool will generate the properties file, which you can then use with the `-silent` option.

- `-checkonly`

Use this flag to check the status of the Oracle software home configuration. Running the command with the `-checkonly` flag does not remove the Oracle configuration.

- `-local`

Use this flag on a multinode environment to deconfigure Oracle software in a cluster.

When you run `deconfig` with this flag, it deconfigures and deinstalls the Oracle software on the local node (the node where `deconfig` is run). On remote nodes, it deconfigures Oracle software, but does not deinstall the Oracle software.

- `-paramfile` *complete path of input parameter property file*

Use this flag to run `deconfig` with a parameter file in a location other than the default. When you use this flag, provide the complete path where the parameter file is located.

The default location of the parameter file depends on the location of `deconfig`:

- From the installation media or stage location: `ORACLE_HOME\response`
- From a unzipped archive file from OTN: `ziplocation\response`
- After installation from the installed Oracle home: `ORACLE_HOME\deinstall\response`

- `-params` [`name1=value name 2=value name3=value . . .`]

Use this flag with a parameter file to override one or more values that you want to change in a parameter file you have already created.

- `-o` *complete path of directory for saving files*

Use this flag to provide a path other than the default location where the properties file is saved. The default location is `\response\deinstall.rsp.tpl`.

The default location of the parameter file depends on the location of `deconfig`:

- From the installation media or stage location before installation: `ORACLE_HOME\`
- From an unzipped archive file from OTN: `\ziplocation\response\`
- After installation from the installed Oracle home: `ORACLE_HOME\deinstall\response`

- `-help` | `-h`

Use the help option (`-help` or `-h`) to obtain additional information about the optional flags

Removing Oracle Software

Complete the following procedure to remove Oracle software:

1. Log in as a member of the Administrators group.
2. Run the `deinstall` command, providing information about the Oracle System Identifier (SID), when prompted.

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 D, "Configuration Worksheet for DRDA"](#)
- [Appendix C, "Initialization Parameters"](#)

Using Response Files for Noninteractive Installation

This appendix describes how to install and configure Oracle products using response files. It includes the following topics:

- [Introduction](#)
- [Using Response Files to Install Oracle Components in Noninteractive Mode](#)
- [Customizing a Sample Response File](#)
- [Creating a New Response File](#)
- [Running Oracle Universal Installer and Specifying a Response File](#)

Introduction

Typically, Oracle Universal Installer runs in interactive mode, which means that it prompts you to provide information in graphical user interface (GUI) screens. Alternatively, you can run Oracle Universal Installer in noninteractive mode. Noninteractive mode is also referred to as silent mode, or silent installation.

You may want to use noninteractive mode to install Oracle Database Gateway on Microsoft Windows (64-bit) in the following scenarios:

- You need to deploy Oracle Components to multiple nodes in an unattended manner. You can schedule the noninteractive installation mode from the operating system scheduler or other job subsystem that your site normally uses.
- No interaction with the user is intended.
- A graphical facility to run Oracle Universal Installer in interactive mode is not available. (Oracle Universal Installer is always available on Microsoft Windows, but not on UNIX systems.)

This section covers the following topics on how you can use response files to run Oracle Universal Installer in noninteractive mode:

Using Response Files to Install Oracle Components in Noninteractive Mode

To use noninteractive mode, you run Oracle Universal Installer with a response file. A response file is a text file that contains variables and values that Oracle Universal Installer uses during the installation process. Oracle provides a set of sample response files that you can customize, or you can create your own response file by recording your installation selections.

See Also: *Oracle Universal Installer and OPatch User's Guide for Windows and UNIX* for more information about response file formats

Customizing a Sample Response File

Oracle provides response file templates for each product and installation type, and for each configuration tool. The response files for Oracle Gateway, `tg.rsp` and `netca.rsp`, are located in the response directory on the media.

Note: If you copied the software to a hard disk, then the response files are located in the `Disk1\response` directory.

To copy and modify a response file:

1. Copy the necessary response files from the `\Response` directory on the installation media to your hard drive.
2. From the Start menu, select **Programs**, then **Oracle - HOME_NAME**, then **Oracle Installation Products**, then **Universal Installer Concepts Guide**.

Oracle Universal Installer and OPatch User's Guide for Windows and UNIX appears in HTML format.

3. Modify the response files with any text file editor by following the instructions in both the response files and *Oracle Universal Installer and OPatch User's Guide for Windows and UNIX*.
4. Run the response file by following the instructions in the "[Running Oracle Universal Installer and Specifying a Response File](#)" section on page A-3.

Creating a New Response File

When you run Oracle Universal Installer in interactive mode, you can record your installation selections into a response file. You do this by running Oracle Universal Installer in Record mode. Oracle Universal Installer generates the response file immediately after you complete the **Summary** page, so you do not need to actually install the gateway to create the response file.

If you want to use the Record mode during a noninteractive installation, then Oracle Universal Installer records the variable values that were specified in the original source response file into the new response file.

Note: You cannot use Record mode to create a response file based on the Basic installation type.

To create a new response file:

1. Make sure that the computer on which you are creating the response file has met the requirements described in respective gateway installation chapters.
2. At the command prompt, use the `cd` command to change to the directory that contains the Oracle Universal Installer `setup.exe` executable.

On the installation media, `setup.exe` is located on Disk 1. If you want to run Oracle Universal Installer from an existing gateway installation, then you can find `setup.exe` in `ORACLE_BASE\ORACLE_HOME\oui\bin`.

3. Enter the following command:

```
setup -record -destinationFile response_file_name
```

Replace *response_file_name* with the complete path for the new response file. For example:

```
setup -record -destinationFile C:\response_files\install_oracle11g
```

4. After Oracle Universal Installer launches, enter the installation settings, which will be recorded into the response file.
5. When the Summary page appears, do one of the following:
 - Click **Install** to continue with the installation.
 - Click **Cancel** if you want to only create the response file but not continue with the installation. The installation will stop, but the settings you have entered will be recorded to the response file.

Later, Oracle Universal Installer will save your new response file using the path and file name you specified on the command line.

6. If necessary, make any environment-specific changes to the response file for the computer on which you will run it.
7. Run the response file by following the instructions in the "[Running Oracle Universal Installer and Specifying a Response File](#)" section.

Running Oracle Universal Installer and Specifying a Response File

You run Oracle Universal Installer at the command line, specifying a response file. The Oracle Universal Installer executable, `setup.exe`, provides several options. For help information on the full set of these options, run `setup.exe` with the `-help` option, for example:

```
C:\ORACLE_BASE\ORACLE_HOME\oui\bin> setup.exe -help
```

To run Oracle Universal Installer and specify a response file:

1. Start a command prompt.
2. Go to the directory where Oracle Universal Installer is installed.
3. From the command line, run Oracle Universal Installer with the correct response file. For example:

```
C:\ORACLE_BASE\ORACLE_HOME\oui\bin> setup.exe [-silent] [-nowelcome]
[-nowait] -responseFile filename
```

Where...	Description
<i>filename</i>	Identifies the full path of the response file
<code>-silent</code>	Runs Oracle Universal Installer in silent mode and suppresses the Welcome screen. If you use <code>-silent</code> , then <code>-nowelcome</code> is not necessary.
<code>-nowelcome</code>	Suppresses the Welcome screen that appears during installation
<code>-nowait</code>	Closes the console window when the silent installation completes

Oracle Database Gateway Troubleshooting

This appendix contains information about troubleshooting. It includes the following topics:

- [Verifying Requirements](#)
- [What to Do if an Installation Error Occurs](#)
- [Reviewing the Log of an Installation Session](#)
- [Troubleshooting Configuration Assistants](#)
- [Noninteractive Installation Response File Error Handling](#)
- [Cleaning Up After a Failed Installation](#)

Verifying Requirements

Before you try any of the troubleshooting steps in this appendix, do the following:

- Check the system requirements section of respective gateway installation chapters to ensure that the system meets the requirements and that you have completed all the preinstallation tasks.
- Read the release notes for the product on your platform before installing it. The release notes are available on the Oracle software installation media. You can find the latest version of the release notes on the Oracle Technology Network Web site:

<http://docs.oracle.com/>

What to Do if an Installation Error Occurs

If you encounter an error during installation then:

- Do not exit Oracle Universal Installer.
- If you clicked **Next** after you entered incorrect information about one of the installation screens, then 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, see the "[Reviewing the Log of an Installation Session](#)" section on page B-2.
- If you encounter an error while a configuration assistant is running, see the "[Troubleshooting Configuration Assistants](#)" section on page B-2.
- If you cannot resolve the problem, then remove the failed installation by following the steps listed in the "[Cleaning Up After a Failed Installation](#)" section on page B-3.

Reviewing the Log of an Installation Session

During an installation, Oracle Universal Installer records all the actions that it performs, in a log file. If you encounter problems during the installation, then review the log file for information about possible causes of the problem.

SYSTEM_DRIVE:\Program Files\Oracle\Inventory\logs

Log file names take the form:

installActionsdate_time.log

For example, if the installation occurred at 9:00:56 A.M. on May 14, 2009, then the log file would be named:

installActions2009-05-14_09-00-56-am.log

Note: Do not delete or manually alter the Inventory directory or its contents. Doing so can prevent Oracle Universal Installer from locating products that you install on your system.

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_BASE\ORACLE_HOME\cfgtoollogs` directory. Try to fix the issue that caused the error.
- If you see the "Fatal Error. Reinstall" message, then look for the cause of the problem by reviewing the log files. Refer to the "Fatal Errors" section on page B-2 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 `installActionsdate_time.log` file.

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 then:

1. Remove the failed installation as described in the "[Cleaning Up After a Failed Installation](#)" section on page B-3.
2. Correct the cause of the fatal error.
3. Reinstall the Oracle software.

Noninteractive Installation Response File Error Handling

To determine whether a noninteractive installation succeeded or failed, check the `installActionsdate_time.log` file, located in `SYSTEM_DRIVE:\Program Files\Oracle\Inventory\logs`.

If necessary, then see the previous section for information about determining the location of the `Inventory` 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 a configuration assistant validates the response file at run time. If the validation fails, then the noninteractive 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. Follow the instructions in [Chapter 17, "Removing Oracle Database Gateway"](#) to run the `deinstall` tool to remove the gateway, remove the Oracle directory, and remove Oracle from the Registry Editor keys. Later, reinstall the 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 user name 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_TIMESTAMP_MAPPING
- HS_FDS_DATE_MAPPING
- HS_FDS_ARRAY_EXEC
- HS_FDS_CONNECT_INFO
- 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
- HS_FDS_QUOTE_IDENTIFIER
- HS_IDLE_TIMEOUT
- HS-NLS_LENGTH_SEMANTICS
- HS_KEEP_REMOTE_COLUMN_SIZE
- HS_FDS_REMOTE_DB_CHARSET
- HS_FDS_SUPPORT_STATISTICS
- HS_FDS_RSET_RETURN_ROWCOUNT
- HS_FDS_SQLLEN_INTERPRETATION
- HS_FDS_REPORT_REAL_AS_DOUBLE

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_TIMESTAMP_MAPPING
- HS_FDS_DATE_MAPPING
- HS_FDS_ARRAY_EXEC
- HS_FDS_CONNECT_INFO
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_FETCH_ROWS
- HS_IDLE_TIMEOUT
- HS-NLS_LENGTH_SEMANTICS
- HS_KEEP_REMOTE_COLUMN_SIZE
- HS_FDS_REMOTE_DB_CHARSET
- HS_FDS_SUPPORT_STATISTICS
- HS_FDS_SQLLEN_INTERPRETATION

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_TIMESTAMP_MAPPING
- HS_FDS_DATE_MAPPING
- HS_FDS_ARRAY_EXEC
- HS_FDS_CONNECT_INFO
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_FETCH_ROWS
- HS_IDLE_TIMEOUT
- HS-NLS_LENGTH_SEMANTICS
- HS_KEEP_REMOTE_COLUMN_SIZE
- HS_FDS_REMOTE_DB_CHARSET
- HS_FDS_SUPPORT_STATISTICS

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 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_PROC_IS_FUNC
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_REPORT_REAL_AS_DOUBLE
- HS_FDS_RESULTSET_SUPPORT
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_FETCH_ROWS
- HS_FDS_TIMESTAMP_MAPPING
- HS_FDS_DATE_MAPPING
- HS_FDS_ARRAY_EXEC
- HS_IDLE_TIMEOUT
- HS_NLS_LENGTH_SEMANTICS
- HS_KEEP_REMOTE_COLUMN_SIZE
- HS_FDS_REMOTE_DB_CHARSET
- HS_FDS_SUPPORT_STATISTICS
- HS_FDS_RSET_RETURN_ROWCOUNT
- HS_FDS_SQLLEN_INTERPRETATION

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_TIMESTAMP_MAPPING
- HS_FDS_DATE_MAPPING
- HS_FDS_ARRAY_EXEC

- HS_FDS_CONNECT_INFO
- HS_FDS_TRACE_LEVEL
- HS_TRANSACTION_MODEL
- HS_FDS_FETCH_ROWS
- HS_FDS_REMOTE_DB_CHARSET
- HS_FDS_SQLLEN_INTERPRETATION
- HS_FDS_REPORT_REAL_AS_DOUBLE

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_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TRANSACTION_MODEL
- HS_FDS_FETCH_ROWS
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_IDLE_TIMEOUT
- HS_FDS_MBCS_TO_GRAPHIC
- HS_FDS_GRAPHIC_TO_MBCS
- HS_FDS_TIMESTAMP_MAPPING
- HS_FDS_DATE_MAPPING
- HS_FDS_ARRAY_EXEC
- HS_FDS_QUOTE_IDENTIFIER
- HS_FDS_ISOLATION_LEVEL
- HS_FDS_PACKAGE_COLLID
- HS_FDS_RECOVERY_ACCOUNT

- [HS_FDS_RECOVERY_PWD](#)
- [HS_NLS_LENGTH_SEMANTICS](#)
- [HS_KEEP_REMOTE_COLUMN_SIZE](#)
- [HS_FDS_RESULTSET_SUPPORT](#)
- [HS_FDS_REMOTE_DB_CHARSET](#)
- [HS_FDS_SUPPORT_STATISTICS](#)
- [HS_FDS_RSET_RETURN_ROWCOUNT](#)
- [HS_FDS_AUTHENTICATE_METHOD](#)
- [HS_FDS_ENCRYPT_SESSION](#)
- [HS_FDS_TRUSTSTORE_FILE](#)
- [HS_FDS_TRUSTSTORE_PASSWORD](#)
- [HS_FDS_SQLLEN_INTERPRETATION](#)

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
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. In almost all cases, `HS_LANGUAGE` should be set exactly the same as Oracle database character set for optimal character set mapping and performance. 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.

As more Oracle databases and non-Oracle databases use Unicode as database character sets, it is preferable to also run the gateway in Unicode character set. To do so, you must set `HS_LANGUAGE=AL32UTF8`. However, when the gateway runs on Windows, the Microsoft ODBC Driver Manager interface can exchange data only in the double-byte character set, UCS2. This results in extra ratio expansion of described buffer and column sizes. Refer to [HS_FDS_REMOTE_DB_CHARSET](#) for instruction on how to adjust to correct sizes.

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.

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 to 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_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_TIME_ZONE

Property	Description
Default value for '+ -]hh:mm'	Derived from the NLS_TERRITORY 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	COMMIT_CONFIRM
Range of Values	COMMIT_CONFIRM, READ_ONLY, READ_ONLY_AUTOCOMMIT, SINGLE_SITE, SINGLE_SITE_AUTOCOMMIT

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.
- **READ_ONLY_AUTOCOMMIT** provides read access to the non-Oracle database that do not have logging. READ_ONLY_AUTOCOMMIT does not apply to Oracle Database Gateway for ODBC.
- **SINGLE_SITE** provides read and write access to the non-Oracle database. However, the gateway cannot participate in distributed updates.
- **SINGLE_SITE_AUTOCOMMIT** provides read and write access to the non-Oracle database which do not have logging. Any update is committed immediately, and the gateway cannot participate in distributed updates. SINGLE_SITE_AUTOCOMMIT does not apply to Oracle Database Gateway for ODBC.

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 do 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 that describes the connection to the non-Oracle system.

The default initialization parameter file already has an entry for this parameter. This release of gateway can support IPv6. If IPv6 address format is to be specified, you would need to wrap square brackets around the IPv6 specification to indicate the separation from the port number. 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/[instance_name] [/database_name]
```

where, *host_name* is the host name or IP address of the machine hosting the SQL Server database, *instance_name* is the instance of SQL Server running on the machine, and *database_name* is the SQL Server database name. Both *instance_name* and *database_name* are optional. If *instance_name* is omitted and *database_name* is provided, the slash (/) is required. This can be shown as follows:

```
HS_FDS_CONNECT_INFO= host_name//database_name
```

For Oracle Database Gateway for ODBC:

```
HS_FDS_CONNECT_INFO=dsn_value
```

where *dsn_value* is the name of the system DSN defined in the Microsoft Windows ODBC Data Source Administrator.

For Oracle Database Gateway for DRDA:

`HS_FDS_CONNECT_INFO=IP_address:Port_number/Database_name,Type`

Where `IP_address` is the hostname or ip address of the DB2 DRDA server

`Port_number` is the port number of the DB2 DRDA server.

`Database_name` is the database name of the DB2 server

`Type` (case insensitive) is one of the following:

- ZOS (DB2 UDB for z/OS),
- IOS (DB2 UDB for iSeries), or
- LUW (DB2 UDB for Linux, UNIX, or Windows)

For example,

`HS_FDS_CONNECT_INFO=[2001:0db8:20C:F1FF:FEC6:38AF]:1300/DB2M,ZOS`

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.

For DRDA, `HS_FDS_RECOVERY_ACCOUNT` 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 `HS_FDS_RECOVERY_ACCOUNT`. If this parameter is missing, then the gateway attempts to connect to a user ID of `RECOVER`.

The name of the recovery account is case-sensitive.

HS_FDS_RECOVERY_PWD

Property	Description
Default Value	none
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.

`HS_FDS_RECOVERY_PWD` is used with the `HS_FDS_RECOVERY_ACCOUNT`. The recovery user connects to the non-Oracle database if a distributed transaction is in doubt.

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_REPORT_REAL_AS_DOUBLE

Property	Description
Default Value	FALSE
Range of Values	TRUE, FALSE

Enables Oracle Database Gateway for SQL Server, Oracle Database Gateway for ODBC, and Oracle Database Gateway for Sybase 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=num

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.

HS_FDS_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	HS_FDS_CAPABILITY= {FUNCTION/{ON OFF SKIP}},...

If the HS_FDS_CAPABILITY is set to ON then the specified function will be sent to DB2 for processing. In other words, post processing will be not needed for that function.

If the HS_FDS_CAPABILITY is set to OFF then the specified function will be not be sent to DB2 for processing. In other words, it will be post processed.

If the `HS_FDS_CAPABILITY` is set to `SKIP` then the specified function will be stripped from the SQL statement sent to DB2. In other words the function will be ignored.

HS_FDS_ISOLATION_LEVEL

Property	Description
Default Value	CHG for DB2 UDB for iSeries, CS for DB2 UDB for z/OS, DB2/UDB
Range of Values	{CHG CS RR ALL NC}
Syntax	HS_FDS_ISOLATION_LEVEL={CHG CS RR ALL NC}

`HS_FDS_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 UDB for iSeries)
CS	Cursor Stability (default for DB2 UDB for Linux, UNIX, and Windows, and DB2 UDB for z/OS)
RR	Repeatable Read
ALL	ALL
NC	No Commit

HS_FDS_PACKAGE_COLLID

Property	Description
Default Value	ORACLEGTW
Range of Values	An alphanumeric string 1 to 18 characters in length
Syntax	HS_FDS_PACKAGE_COLLID= <i>collection_id</i>

`HS_FDS_PACKAGE_COLLID` specifies the package collection ID. Note that in DB2 UDB for iSeries, the collection ID is actually the name of an AS/400 library.

Note: Any change to this parameter will cause a new package to be implicitly bound by the gateway. For DB2 for UDB iSeries, prior to attempting a connection, one should use the iSeries SQL command `CREATE SCHEMA` or `CREATE COLLECTION` to create an iSeries library with the name as specified for `HS_FDS_PACKAGE_COLLID`. This `COLLECTION` or `SCHEMA` should be created under the id specified in the `CONNECT TO` phrase of the Oracle SQL command `CREATE DATABASE LINK`.

HS_IDLE_TIMEOUT

Property	Description
Default Value	0 (no timeout)
Range of Values	0-9999 (minutes)
Syntax	<code>HS_IDLE_TIMEOUT=num</code>

This feature is only available for Oracle Net TCP protocol.

When there is no activity for a connected gateway session for this specified time period, the gateway session would be terminated automatically with pending update (if any) rolled back.

HS_FDS_MBCS_TO_GRAPHIC

Property	Description
Default Value	FALSE
Range of Values	FALSE TRUE
Syntax	<code>HS_FDS_MBCS_TO_GRAPHIC={FALSE TRUE}</code>

If set to `TRUE`, any single-byte character meant to insert to DB2 (var)graphic column would be converted to equivalent double-byte value before the insert operation.

HS_FDS_GRAPHIC_TO_MBCS

Property	Description
Default Value	FALSE
Range of Values	FALSE TRUE
Syntax	<code>HS_FDS_GRAPHIC_TO_MBCS={FALSE TRUE}</code>

If set to `TRUE`, any double-byte characters in DB2 (var)graphic column that can have equivalent single-byte equivalent would be translated to equivalent single-byte before sending to the user.

HS_FDS_TIMESTAMP_MAPPING

Property	Description
Default Value	DATE (except for DB2 which uses CHAR as default)
Range of Values	CHAR DATE TIMESTAMP
Syntax	HS_FDS_TIMESTAMP_MAPPING= {CHAR DATE TIMESTAMP}

If set to CHAR, then non-oracle target timestamp would be mapped to CHAR(26). If set to DATE, then non-Oracle target timestamp would be mapped to Oracle date. If set to TIMESTAMP, then non-Oracle target timestamp would be mapped to Oracle timestamp.

HS_FDS_DATE_MAPPING

Property	Description
Default Value	DATE (except for Teradata which uses CHAR as default)
Range of Values	DATE CHAR
Syntax	HS_FDS_DATE_MAPPING= {DATE CHAR}

If set to CHAR, then non-oracle target date would be mapped to CHAR(10). If set to DATE, then non-Oracle target date would be mapped to Oracle date.

HS_FDS_ARRAY_EXEC

Property	Description
Default Value	TRUE (except for Oracle Database Gateway for DRDA which uses FALSE as default)
Range of values	{TRUE FALSE}
Syntax	HS_FDS_ARRAY_EXEC= {TRUE FALSE}

If set to TRUE, the gateway will use array operations for insert, update, delete statements containing binds against the remote data source. The array size is determined by the value of the HS_FDS_FETCH_ROWS init parameter.

If set to FALSE, the gateway will not use array operations for insert, update, and delete statements. Instead, a single statement will be issued for every value.

HS_FDS_QUOTE_IDENTIFIER

Property	Description
Default Value	TRUE for Sybase
Range of Values	TRUE FALSE
Syntax	HS_FDS_QUOTE_IDENTIFIER= {FALSE TRUE}

By default, the gateway will quote identifiers if the FDS supports it. However, we give the user the ability to overwrite the behavior. `HS_FDS_QUOTE_IDENTIFIER` overrides the target's ability to support quote identifier depending on the value provided.

HS_NLS_LENGTH_SEMANTICS

Property	Description
Default Value	BYTE
Range of values	BYTE CHAR
Syntax	HS_NLS_LENGTH_SEMANTICS = { BYTE CHAR }

This release of gateway has Character Semantics functionality equivalent to the Oracle database Character Semantics, that is, `NLS_LENGTH_SEMANTICS`. When `HS_NLS_LENGTH_SEMANTICS` is set to `CHAR`, the `(VAR)CHAR` columns of non-Oracle database are to be interpreted as having `CHAR` semantics. The only situation the gateway does not honor the `HS_NLS_LENGTH_SEMANTICS=CHAR` setting is when both Oracle and gateway are on the same multi-byte character set.

HS_KEEP_REMOTE_COLUMN_SIZE

Property	Description
Default Value	OFF
Range of Values	OFF LOCAL REMOTE ALL
Syntax	HS_KEEP_REMOTE_COLUMN_SIZE = OFF LOCAL REMOTE ALL
Parameter type	String

`HS_KEEP_REMOTE_COLUMN_SIZE` specifies whether to suppress ratio expansion when computing the length of `(VAR)CHAR` datatypes during data conversion from non-Oracle database to Oracle database. When it is set to `REMOTE`, the expansion is suppressed between the non-Oracle database to the gateway. When it is set to `LOCAL`, the expansion is suppressed between the gateway and Oracle database. When it is set to `ALL`, the expansion is suppressed from the non-Oracle database to the Oracle database.

When the parameter is set, the expansion is suppressed when reporting the remote column size, calculating the implicit resulting buffer size, and instantiating in the local Oracle database. If the gateway runs on Windows and `HS_LANGUAGE=AL32UTF8`, then you must not specify this parameter, as it would influence other ratio related parameter operation. This has effect only for remote column size from non-Oracle database to Oracle database. It has no effect for calculating ratio for data moving from Oracle database to non-Oracle database through gateway during `INSERT`, `UPDATE`, or `DELETE`.

HS_FDS_REMOTE_DB_CHARSET

Property	Description
Default Value	None
Range of values	Not Applicable

Property	Description
Syntax	HS_FDS_REMOTE_DB_CHARSET

This parameter is valid only when `HS_LANGUAGE` is set to `AL32UTF8` and the gateway runs on Windows. As more Oracle databases and non-Oracle databases use Unicode as database character sets, it is preferable to also run the gateway in Unicode character set. To do so, you must set `HS_LANGUAGE=AL32UTF8`. However, when the gateway runs on Windows, the Microsoft ODBC Driver Manager interface can exchange data only in the double-byte character set, UCS2. This results in extra ratio expansion of described buffer and column sizes. To compensate, the gateway can re-adjust the column size if `HS_FDS_REMOTE_DB_CHARSET` is set to the corresponding non-Oracle database character set. For example, `HS_FDS_REMOTE_DB_CHARSET=KO16KSC5601`.

HS_FDS_SUPPORT_STATISTICS

Property	Description
Default Value	TRUE
Range of values	{TRUE FALSE}
Syntax	HS_FDS_SUPPORT_STATISTICS= {TRUE FALSE}

We gather statistics from the non-Oracle database by default. You can choose to disable the gathering of remote database statistics by setting the `HS_FDS_SUPPORT_STATISTICS` parameter to `FALSE`.

HS_FDS_RSET_RETURN_ROWCOUNT

Property	Description
Default Value	FALSE
Range of values	{TRUE FALSE}
Syntax	HS_FDS_RSET_RETURN_ROWCOUNT= {TRUE FALSE}

When set to `TRUE`, the gateway returns the row counts of DML statements that are executed inside a stored procedure. The row count is returned as a single row, single column result set of type signed integer.

When set to `FALSE`, the gateway skips the row counts of DML statements that are executed inside a stored procedure. This is the default behavior, and it is the behavior of 11.1 and older gateways.

HS_FDS_SQLLEN_INTERPRETATION

Property	Description
Default Value	64
Range of values	{64 32}
Syntax	HS_FDS_SQLLEN_INTERPRETATION= {64 32}

This parameter is only valid for 64 bit platforms. ODBC standard specifies `SQLLEN` (of internal ODBC construct) being 64 bit on 64 bit platforms, but some ODBC driver managers and drivers violate this convention, and implement it as 32 bit. In order for the gateway to compensate their behavior, you need to specify `HS_FDS_SQLLEN_INTERPRETATION=32` if you use these types of driver managers and driver.

HS_FDS_AUTHENTICATE_METHOD

Property	Description
Default Value	CLEARTEXT
Range of values	{CLEARTEXT ENCRYPT ENCRYPT_BOTH CLIENT KERBEROS}
Syntax	HS_FDS_AUTHENTICATE_METHOD= {CLEARTEXT ENCRYPT ENCRYPT_BOTH CLIENT KERBEROS}

Specifies the way in which user ID and password are sent to the remote DB2 server and authenticated. Valid values are:

- CLEARTEXT : user ID and password are sent in clear text to server (default).
- ENCRYPT : password is sent encrypted to server.
- ENCRYPT_BOTH : user ID and password are sent encrypted to server.
- CLIENT : user ID is validated on the client side instead of by the server.
- KERBEROS : uses Kerberos to authenticate user ID.

HS_FDS_ENCRYPT_SESSION

Property	Description
Default Value	NONE
Range of values	NONE SSL DB2}
Syntax	HS_FDS_ENCRYPT_SESSION = {NONE SSL DB2}

Specifies the way the session to DB2 is encrypted. Valid values are:

- NONE : data session is not encrypted (default).
- SSL : Use SSL to encrypt data session (supported only by DB2 for iSeries).
- DB2 : Use DB2 encryption protocol for data session (supported only by DB2 for LUW and DB2 for z/OS, and can be used only when authentication is CLEARTEXT, ENCRYPT, or ENCRYPT_BOTH).

HS_FDS_TRUSTSTORE_FILE

Property	Description
Default Value	none
Range of values	<i>path to truststore file</i>
Syntax	HS_FDS_TRUSTSTORE_FILE = <i>path to truststore file</i>

Specifies the path that specifies the location of the truststore file. The truststore file contains a list of the valid Certificate Authorities (CAs) that are trusted by the client machine for SSL server authentication.

HS_FDS_TRUSTSTORE_PASSWORD

Property	Description
Default Value	none
Range of values	<i>password</i>
Syntax	HS_FDS_TRUSTSTORE_PASSWORD= <i>password</i>

Specifies the password required to access the truststore.

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	<ul style="list-style-type: none"> ▪ ORACLE_HOME 	
System ID of the gateway	<ul style="list-style-type: none"> ▪ ORACLE_SID 	
Configuring TCP/IP	<ul style="list-style-type: none"> ▪ Local Host name, Domain Name ▪ IP Address ▪ Network Mask ▪ Name Server IP Address ▪ DRDA server Host name or IP Address ▪ DRDA server Service Port Number 	
Recovery user ID	<ul style="list-style-type: none"> ▪ HS_FDS_RECOVERY_ACCOUNT 	
Recovery Password	<ul style="list-style-type: none"> ▪ HS_FDS_RECOVERY_PWD 	
Remote Database Name	<ul style="list-style-type: none"> ▪ DRDA_REMOTE_DB_NAME 	
Connection Parameter	<ul style="list-style-type: none"> ▪ DRDA_CONNECT_PARM 	
Remote collection ID	HS_FDS_PACKAGE_COLLID	
Remote package name	DRDA_PACKAGE_NAME	
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 following privileges on the remote database; your database administrator will need to provide these.

- package privileges of BIND, COPY, and EXECUTE
 - collection privilege of CREATE IN
 - system privileges of BINDADD and BINDAGENT
-
-

Index

A

action items, 2-1, 4-1, 6-1, 8-1, 10-1, 12-1
application
 authenticating logons, 15-1
AS/400
 command DSPRDBDIRE, 13-3
 defining user ID, 13-3
 library name, HS_FDS_PACKAGE_
 COLLID, C-18

B

Basic installation method
 noninteractive installations, A-2
Bind Package Stored Procedure
 DB2 UDB for iSeries, 13-3
 DB2/OS390, 13-1
bind privilege
 configuration worksheet, D-2
 DB2 UDB for iSeries, 13-3
 DB2 UDB for Linux, UNIX, and Windows, 13-4
 DB2/OS390, 13-1
BINDADD privilege
 configuration worksheet, D-2
 DB2 UDB for Linux, UNIX, and Windows, 13-4
 DB2/OS390, 13-2
BINDAGENT privilege
 configuration worksheet, D-2
 DB2/OS390, 13-2
binding the DRDA package
 authority of user ID and password
 DB2 UDB for iSeries, 13-3
 DB2/OS390, 13-1

C

character sets
 Heterogeneous Services, C-10
checklist
 DRDA server configuration, 13-1
closing and opening again any session against db2
 required with any change to
 HS_FDS_PACKAGE_COLLID, C-19
collection privilege - CREATE IN
 configuration worksheet, D-2

 DB2/OS390, 13-2
collection privilege - CREATETAB,
 DB2/OS390, 13-2
Communication Database (CDB) tables, DDF, 13-2
configuration assistants
 troubleshooting, B-2
configuration assistants, troubleshooting, B-2
Configuring
 two-phase commit, 3-6, 5-6, 7-6, 9-7
configuring
 checklists for DRDA server, 13-1
 DB2 UDB for iSeries, 13-3
 DB2 UDB for Linux, UNIX, and Windows, 13-3
 DB2/OS390, 13-1
 list of parameters needed to configure the
 gateway, D-1
Configuring the gateway, 3-1, 5-1, 7-1, 9-1, 14-1
CONNECT authority
 DB2 UDB for Linux, UNIX, and Windows, 13-4
connect_descriptor, 3-5, 5-5, 7-5, 9-5, 11-6, 14-6
COPY
 privilege
 configuration worksheet, D-2
 DB2/OS390, 13-1
CREATE IN privilege
 configuration worksheet, D-2
 DB2/OS390, 13-2
CREATEIN privilege, DB2 UDB for Linux, UNIX, and
 Windows, 13-4
CREATETAB privilege
 DB2 UDB for Linux, UNIX, and Windows, 13-4
 DB2/OS390, 13-2
Creating
 transaction log table, 3-7, 5-7, 7-7, 9-8
cursor
 stability, HS_FDS_ISOLATION_LEVEL, C-18

D

data dictionary
 support, 14-1
database
 authorities - CONNECT, BINDADD, and
 CREATETAB, 13-4
 link
 defining and controlling, 15-2

- native tool, 14-1
- Database Configuration Assistant (DBCA)
 - troubleshooting, B-2
- Database link
 - behavior, 3-6, 11-6
- database link
 - behavior, 5-6, 7-6, 9-6
- DB2
 - Distributed Data Facility (DDF), 13-2
 - SPUFI utility, 13-2
- DB2 UDB for iSeries
 - configuring the DRDA server, 13-3
 - defining user ID, 13-3
 - HS_FDS_ISOLATION_LEVEL, C-18
 - HS_FDS_PACKAGE_COLLID, C-18
 - user ID mapping, 15-3
- DB2 UDB for Linux, Unix, and Window
 - with SPUFI, 14-1
- DB2 UDB for Linux, UNIX, and Windows
 - configuring, 13-3
 - configuring the DRDA server, 13-3
- DB2 UDB for Linux, Unix, and Windows
 - HS_FDS_ISOLATION_LEVEL, C-18
 - user ID mapping, 15-4
- DB2 UDB for z/OS
 - HS_FDS_ISOLATION_LEVEL, C-18
- DB2/OS390
 - configuring, 13-1
 - user ID mapping, 15-3
- describe cache high water mark
 - definition, C-9
- disk space
 - checking, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
- distributed
 - operations, DB2, 13-2
 - transaction, HS_FDS_RECOVERY_ACCOUNT, C-15
- DRDA
 - session security options, 15-3
- DRDA server
 - configuring
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-3
 - DB2/OS390, 13-1
 - Hostname or IP Address (configuring TCP/IP, worksheet), D-1
 - Service Port Number (configuring TCP/IP, worksheet), D-1
- DRDA Server, Oracle Database Gateway, 12-1
- DRDA, Oracle Database Gateway, 12-1
- DSPRDBDIRE command, 13-3

E

- environment variables
 - TEMP and TMP, hardware requirements, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
- error
 - obsolete parameters, 16-2

- Error messages
 - error tracing, C-16
- errors
 - configuration assistants, B-2
 - installation, B-2
 - noninteractive installation, B-3
- EXECUTE privilege
 - configuration worksheet, D-2
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-1

F

- fatal errors, B-2
- fetch array size, with HS_FDS_FETCH_ROWS, C-17
- files
 - Oracle Universal Installer log files, B-2

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, 3-6, 5-6, 7-7, 9-7
- gateway
 - authenticating logons, 15-1
- Gateway Password Encryption Tool, 3-8, 5-8, 7-8, 9-9, 11-7
- globalization support
 - Heterogeneous Services, C-10

H

- Heterogeneous Services
 - defining maximum number of open cursors, C-11
 - initialization parameters, 11-1
 - optimizing data transfer, C-11
 - Oracle Database Gateway for ODBC
 - creating initialization file, 11-1
 - setting global name, C-9
 - specifying cache high water mark, C-9
 - tuning internal data buffering, C-12
 - tuning LONG data transfer, C-11
 - HS_CALL_NAME initialization parameter, C-8
 - HS_DB_NAME initialization parameter, C-9
 - HS_DESCRIBE_CACHE_HWM initialization parameter, C-9
 - HS_FDS_CONNECT_INFO, C-14
 - HS_FDS_CONNECT_INFO initialization parameter
 - specifying connection information, 11-2
 - HS_FDS_FETCH_ROWS parameter, C-17
 - HS_FDS_ISOLATION_LEVEL parameter, C-18
 - HS_FDS_PACKAGE_COLLID parameter
 - defined, C-18
 - HS_FDS_PROC_IS_FUNC initialization parameter, C-15
 - HS_FDS_RECOVERY_ACCOUNT parameter
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
 - HS_FDS_RECOVERY_PWD initialization

- parameter, C-17
- HS_FDS_RECOVERY_PWD parameter
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2, 13-3
- HS_FDS_RESULTSET_SUPPORT initialization parameter, C-16
- HS_FDS_SHAREABLE_NAME initialization parameter, C-12
- HS_FDS_TRACE_LEVEL initialization parameter, C-16
 - enabling agent tracing, C-2
- HS_FDS_TRANSACTION_LOG initialization parameter, C-17
- HS_KEEP_REMOTE_COLUMN_SIZE initialization parameter, C-21
- HS_LANGUAGE initialization parameter, C-10
- HS_LONG_PIECE_TRANSFER_SIZE initialization parameter, C-11
- HS_OPEN_CURSORS initialization parameter, C-11
- HS_RPC_FETCH_REBLOCKING initialization parameter, C-11
- HS_RPC_FETCH_SIZE initialization parameter, C-12
- HS_TIME_ZONE initialization parameter, C-12
- HS_TRANSACTION_LOG, 3-7, 5-7, 7-7, 9-8
- HS_TRANSACTION_LOG table
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-4

I

- IFILE initialization parameter, C-13
- inbound connections
 - processing, 15-2
- Informix Server, Oracle Database Gateway, 4-1
- Informix, Oracle Database Gateway, 4-1
- 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
- installActions.log file, B-2
- installation
 - accessing installation software, 1-3, 1-4
 - checklists
 - DRDA server, 13-1
 - downloading software from Oracle Technology Network, 1-3
 - errors
 - log session, B-2
 - while configuration assistant runs, B-2
 - log files, B-2
 - noninteractive error handling, B-3
 - procedure, 1-4
 - response files, B-3
 - reviewing a log of an installation session, B-2
 - troubleshooting, B-1, B-3
- installation software, accessing, 1-3, 1-4

- installations
 - log file, B-2
- isolation level, HS_FDS_ISOLATION_LEVEL, C-18

L

- listener, 3-5, 5-5, 7-5, 9-5, 11-6, 14-6
- listener.ora file, 3-10, 5-9, 7-9, 9-10, 11-8, 14-8
 - example, 3-3, 5-3, 7-3, 9-3, 11-3, 14-3
 - location, 3-2, 5-2, 7-2, 9-3, 11-3, 14-2
- log files, B-2
 - reviewing an installation session, B-2
 - troubleshooting, B-2

N

- Net Configuration Assistant, troubleshooting, B-2
- noninteractive installation
 - errors, B-3
 - Record mode, A-2
 - See also* noninteractive deinstallation, response files, A-1

O

- obsolete parameters since V4 gateway, 16-2
- ODBC connectivity
 - specifying path to library, C-12
- ODBC, Oracle Database Gateway, 10-1
- option
 - DRDA session security, 15-3
 - security conduct, 15-2
- Oracle Database Gateway
 - DRDA, 12-1
 - DRDA Server, 12-1
 - Informix, 4-1
 - Informix Server, 4-1
 - ODBC, 10-1
 - SQL Server, 8-1
 - Sybase, 2-1
 - Sybase Server, 2-1
 - Teradata Server, 6-1
- Oracle Database Gateway for ODBC
 - creating initialization file, 11-1
- Oracle Net
 - configuring, 3-2, 5-2, 7-2, 9-2, 11-3
 - operating system authentication, 15-2
- Oracle Net Listener, 3-5, 5-5, 7-5, 9-5, 11-6, 14-6
 - starting, 3-4, 5-4, 7-4, 9-4, 11-4, 14-4
- Oracle Technology Network (OTN)
 - accessing, 1-3
 - downloading software from, 1-3
- Oracle Universal Installer (OUI)
 - log files, B-2
- ore, D-1
- OTN. *See* Oracle Technology Network, 1-3

P

- package
 - collection id, HS_FDS_PACKAGE_COLLID, C-18

- privileges - BIND and EXECUTE, DB2 UDB for Linux, UNIX, and Windows, 13-4
- privileges - BIND, COPY, and EXECUTE configuration worksheet, D-2
 - DB2/OS390, 13-1
- parameter
 - list of parameters needed to configure the gateway, D-1
 - obsolete since V4 gateway, 16-2
- parameters
 - gateway initialization file
 - HS_FDS_CAPABILITY, C-17
 - HS_FDS_FETCH_ROWS, C-17
 - HS_FDS_ISOLATION_LEVEL, C-18
 - HS_FDS_PACKAGE_COLLID, C-18
 - HS_FDS_RECOVERY_ACCOUNT
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
 - HS_FDS_RECOVERY_PWD
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
- privileges
 - BIND
 - configuration worksheet, D-2
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-1
 - BINDADD
 - configuration worksheet, D-2
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
 - BINDAGENT
 - configuration worksheet, D-2
 - DB2/OS390, 13-2
 - CONNECT
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - COPY
 - configuration worksheet, D-2
 - DB2/OS390, 13-1
 - CREATE IN
 - configuration worksheet, D-2
 - DB2/OS390, 13-2
 - CREATEIN
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - CREATETAB
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
 - EXECUTE
 - configuration worksheet, D-2
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-1

R

- RECOVER user ID
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
 - HS_FDS_RECOVERY_ACCOUNT, C-15
- recovery user ID and password
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
- RECOVERY_ACCOUNT
 - account user name, 3-7, 5-7, 7-7, 9-7
 - creating a recovery account, 3-7, 5-7, 7-7, 9-7
- remote
 - database
 - configuration worksheet, D-2
 - DB2 UDB for iSeries, 13-3
 - DB2 UDB for Linux, UNIX, and Windows, 13-4
 - DB2/OS390, 13-2
 - DRDA database, HS_FDS_ISOLATION_LEVEL, C-18
- remote functions
 - referenced in SQL statements, C-8
- response files
 - creating, A-2
 - customizing, A-2
 - samples, A-2
 - specifying during installation, A-3
 - See also* noninteractive installation, A-1

S

- schema privileges - CREATEIN, 13-4
- security
 - overview, 15-1
- SID, 3-1, 5-1, 7-1, 9-1, 11-1
- silent installation. *See* noninteractive installation, A-1
- SPUFI on DB2/OS390, 14-1
- SQL
 - statements, HS_FDS_ISOLATION_LEVEL, C-18
- SQL Server, Oracle Database Gateway, 8-1
- SQL Server, Oracle Database Gateway, 8-1
- stability, of cursor, HS_FDS_ISOLATION_LEVEL, C-18
- Sybase Server, Oracle Database Gateway, 2-1
- Sybase, Oracle Database Gateway, 2-1
- system privileges - BINDADD and BINDAGENT
 - configuration worksheet, D-2
 - DB2/OS390, 13-2

T

- TEMP
 - environment variable, hardware requirements, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
- temporary directory, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
- temporary disk space
 - checking, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2

- freeing, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
- Teradata Server, Oracle Database Gateway, 6-1
- tmp directory
 - checking space in, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
 - freeing space in, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
- TMP environment variable, hardware
 - requirements, 2-2, 4-2, 6-2, 8-2, 10-2, 12-2
- tnsnames.ora, 3-5, 5-5, 7-5, 9-5, 11-6, 14-6
 - configuring, 3-5, 5-5, 7-5, 9-5, 11-6, 14-6
 - multiple listeners, 3-5, 5-5, 7-5, 9-5, 11-6, 14-6
- Transaction log table
 - creating, 3-7, 5-7, 7-7, 9-8
- troubleshooting, B-1, B-3
 - fatal errors, B-2
 - Inventory log files, B-2
- Two-phase commit
 - configuration, 3-6, 5-6, 7-7, 9-7
 - transaction log table, 3-7, 5-7, 7-7, 9-8
- two-phase commit
 - HS_TRANSACTION_LOG table
 - DB2 UDB for iSeries, 13-3

U

- user ID mapping
 - DB2 UDB for iSeries, 15-3
 - DB2/OS390, 15-3

