Preface

Intended Audience

Documentation Accessibility

Related Documents

Conventions

Part I

Overview of the Oracle Database Gateway Installation

1

Overview of the Oracle Database Gateway Installation

Gateway Installation Configurations

Gateway Installation Methods

Interactive Installation Method

Automated Installation Method Using Response Files

Installation Considerations

Release Notes

Hardware and Software Certification

Multiple Oracle Homes Support

Upgrades

Accessing the Installation Software

Downloading Oracle Software from the OTN Web Site

Copying the Oracle Software to a Hard Disk

Running the Oracle Universal Installer

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

Hardware Requirements

Software Requirements

Step Through the Oracle Universal Installer

3

Configuring Oracle Database Gateway for Sybase

Configure the Gateway Initialization Parameter File
### Part III Installing and Configuring Oracle Database Gateway for Informix

#### 4 Installing Oracle Database Gateway for Informix

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Requirements for Oracle Database Gateway for Informix</td>
<td>4-1</td>
</tr>
<tr>
<td>Hardware Requirements</td>
<td>4-1</td>
</tr>
<tr>
<td>Software Requirements</td>
<td>4-2</td>
</tr>
<tr>
<td>Step Through the Oracle Universal Installer</td>
<td>4-3</td>
</tr>
</tbody>
</table>

#### 5 Configuring Oracle Database Gateway for Informix

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure the Gateway Initialization Parameter File</td>
<td>5-1</td>
</tr>
<tr>
<td>Choose a System Identifier for the Gateway</td>
<td>5-1</td>
</tr>
<tr>
<td>Customize the Initialization Parameter File</td>
<td>5-1</td>
</tr>
<tr>
<td>Configure Oracle Net for the Gateway</td>
<td>5-2</td>
</tr>
<tr>
<td>Configure Oracle Net Listener for the Gateway</td>
<td>5-2</td>
</tr>
<tr>
<td>Stop and Start the Oracle Net Listener for the Gateway</td>
<td>5-4</td>
</tr>
<tr>
<td>Configure the Oracle Database for Gateway Access</td>
<td>5-4</td>
</tr>
<tr>
<td>Configuring tnsnames.ora</td>
<td>5-4</td>
</tr>
<tr>
<td>Configuring tnsnames.ora for Multiple Listeners</td>
<td>5-5</td>
</tr>
<tr>
<td>Create Database Links</td>
<td>5-6</td>
</tr>
<tr>
<td>Configure Two-Phase Commit</td>
<td>5-6</td>
</tr>
<tr>
<td>Create a Recovery Account and Password</td>
<td>5-7</td>
</tr>
<tr>
<td>Create the Transaction Log Table</td>
<td>5-7</td>
</tr>
<tr>
<td>Encrypt Gateway Initialization Parameter Values</td>
<td>5-8</td>
</tr>
<tr>
<td>Configure Oracle Net for the Gateway</td>
<td>5-8</td>
</tr>
<tr>
<td>Create the Gateway Initialization Parameter File</td>
<td>5-8</td>
</tr>
</tbody>
</table>
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 .............................................................................. 6-3

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
Configuring tnsnames.ora ...................................................................................................... 11-5
Configuring tnsnames.ora for Multiple Listeners ................................................................. 11-6
Create Database Links ......................................................................................................... 11-6
Encrypt Gateway Initialization Parameter Values ..................................................................... 11-7
Configure the Gateway to Access Multiple ODBC Data Sources .......................................... 11-7
Multiple ODBC Data Sources Example: Configuring the Gateway .................................... 11-7
Part VII Installing and Configuring Oracle Database Gateway for DRDA

12 Installing Oracle Database Gateway for DRDA

System Requirements for Oracle Database Gateway for DRDA ............................................... 12-1
  Hardware Requirements ........................................................................................................ 12-1
  Software Requirements ....................................................................................................... 12-2
Step Through the Oracle Universal Installer ........................................................................... 12-3

13 Configuring the DRDA Server

Configuring the DRDA Server for DB2 UDB for z/OS .......................................................... 13-1
Configuring the DRDA Server for DB2 UDB for iSeries ....................................................... 13-3
Configuring the DRDA Server for DB2 UDB for Linux, UNIX, and Windows ...................... 13-3

14 Configuring Oracle Database Gateway for DRDA

Configure the Gateway Initialization Parameter File ............................................................... 14-1
  Choose a System Identifier for the Gateway ................................................................... 14-1
  Customize the Initialization Parameter File ..................................................................... 14-2
Configure Oracle Net for the Gateway .................................................................................. 14-2
  Configure Oracle Net Listener for the Gateway ......................................................... 14-2
  Stop and Start the Oracle Net Listener for the Gateway ........................................... 14-4
Configure Two-Phase Commit ............................................................................................ 14-4
Create Tables and Views for Data Dictionary Support ......................................................... 14-4
Configure the Oracle Database for Gateway Access ......................................................... 14-5
  Configuring tnsnames.ora ......................................................................................... 14-6
  Configuring tnsnames.ora for Multiple Listeners ..................................................... 14-6
Create Database Links .......................................................................................................... 14-7
Configure the Gateway to Access Multiple DRDA Databases ........................................... 14-7
  Multiple DRDA Databases Example: Configuring the Gateway ................................ 14-8
  Multiple DRDA Databases Example: Configuring Oracle Net Listener ...................... 14-8
  Multiple DRDA Databases Example: Stopping and Starting the Oracle Net Listener ...... 14-9
  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
<table>
<thead>
<tr>
<th>Links and CONNECT Clauses</th>
<th>15-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processing Inbound Connections</td>
<td>15-2</td>
</tr>
<tr>
<td>User ID Mapping</td>
<td>15-3</td>
</tr>
<tr>
<td>Passwords in the Gateway Initialization File</td>
<td>15-4</td>
</tr>
</tbody>
</table>

### 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/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
<table>
<thead>
<tr>
<th>Initialization Parameter Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS_CALL_NAME</td>
<td>C-8</td>
</tr>
<tr>
<td>HS_DB_DOMAIN</td>
<td>C-8</td>
</tr>
<tr>
<td>HS_DB_INTERNAL_NAME</td>
<td>C-8</td>
</tr>
<tr>
<td>HS_DB_NAME</td>
<td>C-9</td>
</tr>
<tr>
<td>HS_DESCRIBE_CACHE_HWM</td>
<td>C-9</td>
</tr>
<tr>
<td>HS_LANGUAGE</td>
<td>C-9</td>
</tr>
<tr>
<td>HS_LONG_PIECE_TRANSFER_SIZE</td>
<td>C-10</td>
</tr>
<tr>
<td>HS_OPEN_CURSORS</td>
<td>C-11</td>
</tr>
<tr>
<td>HS_RPC_FETCH_REBLOCKING</td>
<td>C-11</td>
</tr>
<tr>
<td>HS_RPC_FETCH_SIZE</td>
<td>C-11</td>
</tr>
<tr>
<td>HS_TIME_ZONE</td>
<td>C-12</td>
</tr>
<tr>
<td>HS_TRANSACTION_MODEL</td>
<td>C-12</td>
</tr>
<tr>
<td>IFILE</td>
<td>C-13</td>
</tr>
<tr>
<td>HS_FDS_CONNECT_INFO</td>
<td>C-13</td>
</tr>
<tr>
<td>HS_FDS_DEFAULT_OWNER</td>
<td>C-14</td>
</tr>
<tr>
<td>HS_FDS_PROC_IS_FUNC</td>
<td>C-15</td>
</tr>
<tr>
<td>HS_FDS_RECOVERY_ACCOUNT</td>
<td>C-15</td>
</tr>
<tr>
<td>HS_FDS_RECOVERY_PWD</td>
<td>C-15</td>
</tr>
<tr>
<td>HS_FDS_RESULTSET_SUPPORT</td>
<td>C-16</td>
</tr>
<tr>
<td>HS_FDS_TRACE_LEVEL</td>
<td>C-16</td>
</tr>
<tr>
<td>HS_FDS_TRANSACTION_LOG</td>
<td>C-16</td>
</tr>
<tr>
<td>HS_FDS_REPORT_REAL_AS_DOUBLE</td>
<td>C-17</td>
</tr>
<tr>
<td>HS_FDS_FETCH_ROWS</td>
<td>C-17</td>
</tr>
<tr>
<td>HS_FDS_CAPABILITY</td>
<td>C-17</td>
</tr>
<tr>
<td>HS_FDS_ISOLATION_LEVEL</td>
<td>C-17</td>
</tr>
<tr>
<td>HS_FDS_PACKAGE_COLLID</td>
<td>C-18</td>
</tr>
<tr>
<td>HS_IDLE_TIMEOUT</td>
<td>C-18</td>
</tr>
<tr>
<td>HS_FDS_MBCS_TO_GRAPHIC</td>
<td>C-19</td>
</tr>
<tr>
<td>HS_FDS_GRAPHIC_TO_MBCS</td>
<td>C-19</td>
</tr>
<tr>
<td>HS_FDS_TIMESTAMP_MAPPING</td>
<td>C-19</td>
</tr>
<tr>
<td>HS_FDS_QUOTE_IDENTIFIER</td>
<td>C-19</td>
</tr>
<tr>
<td>HS_NLS_LENGTH_SEMANTICS</td>
<td>C-20</td>
</tr>
<tr>
<td>HS_KEEP_REMOTE_COLUMN_SIZE</td>
<td>C-20</td>
</tr>
<tr>
<td>HS_FDS_REMOTE_DB_CHARSET</td>
<td>C-20</td>
</tr>
<tr>
<td>HS_FDS_SUPPORT_STATISTICS</td>
<td>C-21</td>
</tr>
<tr>
<td>HS_FDS_RSET_RETURN_ROWCOUNT</td>
<td>C-21</td>
</tr>
<tr>
<td>HS_FDS_SQLLEN_INTERPRETATION</td>
<td>C-21</td>
</tr>
<tr>
<td>HS_FDS_AUTHENTICATE_METHOD</td>
<td>C-22</td>
</tr>
<tr>
<td>HS_FDS_ENCRYPT_SESSION</td>
<td>C-22</td>
</tr>
<tr>
<td>HS_FDS_TRUSTSTORE_FILE</td>
<td>C-22</td>
</tr>
<tr>
<td>HS_FDS_TRUSTSTORE_PASSWORD</td>
<td>C-23</td>
</tr>
</tbody>
</table>
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 (32-bit) and 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
Conventions

The following text conventions are used in this manual:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>bold</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary</td>
</tr>
<tr>
<td><em>italics</em></td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>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.</td>
</tr>
<tr>
<td><strong>UPPERCASE</strong></td>
<td>Uppercase letters indicate Structured Query Language (SQL) reserved words, initialization parameters, and environment variables.</td>
</tr>
<tr>
<td>[text]</td>
<td>Brackets are used in syntax statements for optional elements.</td>
</tr>
<tr>
<td>[text</td>
<td>text]</td>
</tr>
<tr>
<td>{text</td>
<td>text}</td>
</tr>
</tbody>
</table>
Part I

Overview of the Oracle Database Gateway Installation

Part I contains the following chapter:

- Chapter 1, "Overview of the Oracle Database Gateway Installation"
This chapter describes the installation of Oracle Database Gateways on Microsoft Windows (32-bit) and 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
- Upgrades
- Accessing the Installation Software
- Running the Oracle Universal Installer

**Gateway Installation Configurations**

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

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

Release Notes

Read the release notes for the product before installing it. The release notes are
available on the Oracle Database 11g Release 2 (11.2) installation media. The latest
version of the release notes is also available on the Oracle Technology Network (OTN)
Web site:

http://www.oracle.com/technetwork/documentation/index.html

Hardware and Software Certification

The platform-specific hardware and software requirements included in this
installation guide were current at the time this guide was published. However,
because new platforms and operating system software versions might be certified after
this guide is published, review the certification matrix on the 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 release 11.2 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.
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 to a Hard Disk

Downloading Oracle Software from the OTN Web Site

You can download the installation files from the OTN and extract them to your hard disk.

To download the installation files:

1. Use any browser to access the OTN software download page:
   
   http://www.oracle.com/technetwork/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 to a Hard Disk

Before installing Oracle Database Gateway, you might want to copy the software to the hard disk. This enables the installation process to run faster. Before copying the installation media content to the hard disk, you must mount the installation media.

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

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 the software from disc, then mount the appropriate disc if it is not already mounted.
   
   If you are installing from a hard disk, 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:

   \( \text{DRIVE}_\text{LETTER}:\autorun\autorun.exe \)

   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.
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"
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.
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 (32-bit) operating systems:

- Microsoft Windows 2000 with Service Pack 1 or later. All editions, including Terminal Services and Microsoft Windows 2000 MultiLanguage Edition (MLE), are supported.

---

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

<table>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total disk space</td>
<td>4 GB</td>
<td>5 GB</td>
</tr>
<tr>
<td>Physical Memory (RAM)</td>
<td>1 GB of real memory is recommended to support the gateway. The total real memory requirement for the concurrent use of the gateway also depends on these factors:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The SQL statement processed by the user</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The number of cursors currently opened against Sybase</td>
<td></td>
</tr>
<tr>
<td></td>
<td>The number of columns in the table being accessed</td>
<td></td>
</tr>
<tr>
<td>Minimum of 1 GB</td>
<td>Double the amount of RAM</td>
<td>Double the amount of RAM</td>
</tr>
<tr>
<td>Virtual memory</td>
<td>256 colors</td>
<td>256 colors</td>
</tr>
<tr>
<td>Video adapter</td>
<td>550 MHz minimum(On Microsoft Windows Vista, 800 MHz minimum)</td>
<td>AMD64, or Intel Extended memory (EM64T)</td>
</tr>
<tr>
<td>Processor</td>
<td>550 MHz minimum(On Microsoft Windows Vista, 800 MHz minimum)</td>
<td>AMD64, or Intel Extended memory (EM64T)</td>
</tr>
</tbody>
</table>
Microsoft Windows Server 2003 - all editions
- Microsoft Windows Server 2003 R2 - all editions
- Microsoft Windows XP Professional
- Microsoft Windows Vista - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008


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:

Step Through the Oracle Universal Installer

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

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Universal Installer: Welcome</td>
<td>Click Next.</td>
</tr>
<tr>
<td>Oracle Universal Installer: Specify Home Details</td>
<td>Specify a name for the installation in the Name field. You can also choose not to edit the default setting of the Name field of the Specify Home Details screen. The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to ORACLE_HOME. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen.</td>
</tr>
</tbody>
</table>
| Oracle Universal Installer: Available Product Components | a. Select Oracle Database Gateway for Sybase 11.2  
|                                                    | 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.                                                |
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 \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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>host_name</code></td>
<td>is the host name or IP address of the machine hosting the Sybase database.</td>
</tr>
<tr>
<td><code>port_number</code></td>
<td>is the port number of the Sybase database server.</td>
</tr>
<tr>
<td><code>database_name</code></td>
<td>is the Sybase database name.</td>
</tr>
</tbody>
</table>

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:

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

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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gateway_sid</td>
<td>specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the tnsnames.ora file.</td>
</tr>
<tr>
<td>oracle_home_directory</td>
<td>specifies the Oracle home directory where the gateway resides.</td>
</tr>
<tr>
<td>dg4sybs</td>
<td>specifies the executable name of the Oracle Database Gateway for Sybase.</td>
</tr>
</tbody>
</table>
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)
  )
```

---

3-4  Oracle Database Gateway Installation and Configuration Guide
Configure the Oracle Database for Gateway Access

```plaintext
(HOST=host_name)
(PORT=port_number)

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

Where:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect_descriptor</td>
<td>is the description of the object to connect to as specified when creating the database link, such as dg4sybs.</td>
</tr>
<tr>
<td>TCP</td>
<td>is the TCP protocol used for TCP/IP connections.</td>
</tr>
<tr>
<td>host_name</td>
<td>specifies the machine where the gateway is running.</td>
</tr>
<tr>
<td>port_number</td>
<td>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 &quot;Syntax of listener.ora File Entries&quot; on page 3-3.</td>
</tr>
<tr>
<td>gateway_sid</td>
<td>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 &quot;Configure Oracle Net Listener for the Gateway&quot; on page 3-2 for more information.</td>
</tr>
<tr>
<td>(HS=OK)</td>
<td>specifies that this connect descriptor connects to a non-Oracle system.</td>
</tr>
</tbody>
</table>

**Configuring tnsnames.ora for Multiple Listeners**

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

```plaintext
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.

Check the sqlnet.ora file for the following parameter setting:

```
names.directory_path = {TNSNAMES}
```

Note: The sqlnet.ora file is typically stored in ORACLE_HOME\network\admin.

TCP

**host_name** specifies the machine where the gateway is running.

**port_number** 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 3-3.

**gateway_sid** 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 3-2 for more information.

**(HS=OK)** specifies that this connect descriptor connects to a non-Oracle system.
Create Database Links

Any Oracle client connected to the Oracle database can access 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
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO user IDENTIFIED BY password USING 'tns_name_entry';
```

Where:

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

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

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

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

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 Administrator’s Guide and Oracle Database Heterogeneous Connectivity User’s Guide for more information about using database links.
Configure Two-Phase Commit

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.
Create Sybase Views for Data Dictionary Support

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.

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:


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.
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:

```bash
> 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 new 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=
```
Configure the Gateway to Access Multiple Sybase Databases

Configuring Oracle Database Gateway for Sybase

3-11

(ADDRESS=
  (PROTOCOL=TCP)
  (PORT=port_number)
  (HOST=host_name))
  (CONNECT_DATA=
    (SID=sg4sybs2))
  (HS=OK))

new_db3_using=
(Description=
  (ADDRESS=
    (PROTOCOL=TCP)
    (PORT=port_number)
    (HOST=host_name))
    (CONNECT_DATA=
      (SID=sg4sybs3))
    (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 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 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" 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"
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 4-3 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 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>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total disk space</td>
<td>4 GB</td>
<td>5 GB</td>
</tr>
</tbody>
</table>
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 (32-Bit) operating systems:

- Microsoft Windows 2000 with Service Pack 1 or later. All editions, including Terminal Services and Microsoft Windows 2000 MultiLanguage Edition (MLE), are supported.
- Microsoft Windows Server 2003 - all editions
Microsoft Windows Server 2003 R2 - all editions
- Microsoft Windows XP Professional
- Microsoft Windows Vista - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008


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:


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

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Universal Installer: Welcome</td>
<td>Click Next.</td>
</tr>
<tr>
<td>Oracle Universal Installer: Specify Home Details</td>
<td>Specify a name for the installation in the Name field. You can also choose not to edit the default setting of the Name field of the Specify Home Details screen. The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to ORACLE_HOME. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen.</td>
</tr>
<tr>
<td>Informix Database Gateway for Informix</td>
<td>b. Click Next.</td>
</tr>
<tr>
<td>Informix Database Server Host Name</td>
<td>Specify the host name of the machine hosting the Informix database server.</td>
</tr>
<tr>
<td>Informix Database Server Port number</td>
<td>Specify the port number of the Informix database server.</td>
</tr>
<tr>
<td>Informix Server Name</td>
<td>Specify the Informix server name</td>
</tr>
<tr>
<td>Informix Database Name</td>
<td>Specify the Informix database name</td>
</tr>
<tr>
<td>Click Next</td>
<td></td>
</tr>
</tbody>
</table>
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.
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:
Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway

Configure Oracle Net for the Gateway
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:

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

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

```
SID_LISTENER=  
  {SID_LIST=
    {SID_DESC=
      {SID_NAME=gateway_sid}
      {ORACLE_HOME=oracle_home_directory}
      {PROGRAM=dg4ifmx}
    }
  }
```

Where:

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

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_NAME=gateway_sid}
      {ORACLE_HOME=oracle_home_directory}
      {PROGRAM=dg4ifmx}
    }
  }
```

Where:

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

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.
Configure the Oracle Database for Gateway Access

```
SID_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

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect_descriptor</td>
<td>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:</td>
</tr>
<tr>
<td></td>
<td>names.directory_path = (TNSNAMES)</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The sqlnet.ora file is typically stored in ORACLE_HOME\network\admin.</td>
</tr>
<tr>
<td>TCP</td>
<td>is the TCP protocol used for TCP/IP connections.</td>
</tr>
<tr>
<td>host_name</td>
<td>specifies the machine where the gateway is running.</td>
</tr>
<tr>
<td>port_number</td>
<td>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 &quot;Syntax of listener.ora File Entries&quot; on page 5-3.</td>
</tr>
<tr>
<td>gateway_sid</td>
<td>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 &quot;Configure Oracle Net Listener for the Gateway&quot; on page 5-2 for more information.</td>
</tr>
<tr>
<td>{HS=OK}</td>
<td>specifies that this connect descriptor connects to a non-Oracle system.</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>dblink</code></td>
<td>is the complete database link name.</td>
</tr>
<tr>
<td><code>tns_name_entry</code></td>
<td>specifies the Oracle Net connect descriptor specified in the tsnames.ora file that identifies the gateway</td>
</tr>
</tbody>
</table>

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**

   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.

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:

```bash
> 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:

```ini
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}
  }
```
Configure the Gateway to Access Multiple Informix Databases

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

SQL> SELECT * FROM ALL_USERS@IFMX3;
```
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"
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 6-3 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 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

<table>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total disk space</td>
<td>4 GB</td>
<td>5 GB</td>
</tr>
</tbody>
</table>
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 (32-bit) operating systems:

- Microsoft Windows 2000 with Service Pack 1 or later. All editions, including Terminal Services and Microsoft Windows 2000 MultiLanguage Edition (MLE), are supported.
- Microsoft Windows Server 2003 - all editions
Microsoft Windows Server 2003 R2 - all editions
Microsoft Windows XP Professional
Microsoft Windows Vista - Business, Enterprise, and Ultimate editions
Microsoft Windows 2008


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:


Step Through the Oracle Universal Installer

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

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Universal Installer: Welcome</td>
<td>Click Next.</td>
</tr>
</tbody>
</table>
| Oracle Universal Installer: Specify Home Details | Specify a name for the installation in the Name field. You can also choose not to edit the default setting of the Name field of the Specify Home Details screen.  
The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to ORACLE_HOME. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen. |
| Oracle 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. |
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.
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:
Configure Oracle Net for the Gateway

$ORACLE_HOME/dg4tera/admin/initdg4tera.ora$

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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$ip_address$</td>
<td>is the IP address of the machine hosting the Teradata database.</td>
</tr>
<tr>
<td>$port_number$</td>
<td>is the port number of the Teradata database server.</td>
</tr>
<tr>
<td>$database_name$</td>
<td>is the Teradata database name. The $database_name$ variable is optional.</td>
</tr>
</tbody>
</table>

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:

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

Where:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>host_name</code></td>
<td>is the name of the machine on which the gateway is installed.</td>
</tr>
<tr>
<td><code>port_number</code></td>
<td>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.</td>
</tr>
</tbody>
</table>

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

```
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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>gateway_sid</code></td>
<td>specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <code>tnsnames.ora</code> file.</td>
</tr>
<tr>
<td><code>oracle_home_directory</code></td>
<td>specifies the Oracle home directory where the gateway resides.</td>
</tr>
<tr>
<td><code>dg4tera</code></td>
<td>specifies the executable name of the Oracle Database Gateway for Teradata.</td>
</tr>
<tr>
<td><code>teradata_client_directory</code></td>
<td>specifies the directory where the Teradata client resides.</td>
</tr>
</tbody>
</table>
Configure the Oracle Database for Gateway Access

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.
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>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect_descriptor</td>
<td>is the description of the object to connect to as specified when creating the database link, such as dg4tera.</td>
</tr>
<tr>
<td>host_name</td>
<td>specifies the machine where the gateway is running.</td>
</tr>
<tr>
<td>port_number</td>
<td>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 7-3.</td>
</tr>
<tr>
<td>gateway_sid</td>
<td>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 7-2 for more information.</td>
</tr>
<tr>
<td>{HS=OK}</td>
<td>specifies that this connect descriptor connects to a non-Oracle system.</td>
</tr>
</tbody>
</table>

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}}}
```

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
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO host_name_1 IDENTIFIED BY password USING 'tns_name_entry';
```

Where:

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

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

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

See Also: Oracle Database Administrator’s Guide for more information about using database links.

Configure Two-Phase Commit

The gateway supports the following transaction capabilities:

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

See Also: *Oracle Database Heterogeneous Connectivity 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
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:

```bash
> 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:

```ini
SID_LIST_LISTENER=
(SID_LIST=
  (SID_DESC=
    (SID_NAME=dg4tera)
    (ORACLE_HOME=oracle_home_directory)
  )
(SID_NAME=dg4tera2)
(SID_NAME=dg4tera3)
```

Configure Oracle Database Gateway for Teradata 7-9
Configure the Gateway to Access Multiple Teradata Databases

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 Teardata 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))
Configure the Gateway to Access Multiple Teradata Databases

Configuring Oracle Database Gateway for Teradata

```
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 '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 '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" describes how to install and configure 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"
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>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total disk space</td>
<td>4 GB</td>
<td>5 GB</td>
</tr>
</tbody>
</table>
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 (32-bit) operating systems:

- Microsoft Windows 2000 with Service Pack 1 or later. All editions, including Terminal Services and Microsoft Windows 2000 MultiLanguage Edition (MLE), are supported.
- Microsoft Windows Server 2003 - all editions

<table>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
</table>
| Physical Memory   | 1 GB of real memory is recommended to support the gateway. The total real memory requirement for the concurrent use of the gateway also depends on these factors:  
  ■ The SQL statement processed by the user  
  ■ The number of cursors currently opened against SQL Server  
  ■ The number of columns in the table being accessed | Minimum of 1 GB                                      |
| Virtual memory    | Double the amount of RAM                                                                      | Double the amount of RAM                             |
| Video adapter     | 256 colors                                                                                   | 256 colors                                           |
| Processor         | 550 MHz minimum (On Microsoft Windows Vista, 800 MHz minimum)                                | AMD64, or Intel Extended memory (EM64T)              |
Microsoft Windows Server 2003 R2 - all editions
- Microsoft Windows XP Professional
- Microsoft Windows Vista - Business, Enterprise, and Ultimate editions
- Microsoft Windows 2008


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:


---

### Step Through the Oracle Universal Installer

**Table 8–2 The Oracle Universal Installer: Steps for Installing the Gateway**

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Universal Installer: Welcome</td>
<td>Click Next.</td>
</tr>
<tr>
<td>Oracle Universal Installer: Specify Home Details</td>
<td>Specify a name for the installation in the Name field. You can also choose not to edit the default setting of the Name field of the Specify Home Details screen. The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to ORACLE_HOME. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen.</td>
</tr>
<tr>
<td>Oracle Database Gateway for SQL Server</td>
<td>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.</td>
</tr>
</tbody>
</table>
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.

---

**Table 8–2 (Cont.) The Oracle Universal Installer: Steps for Installing the Gateway**

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Universal Installer: Summary</td>
<td>The Installation Summary screen enables you to review a tree list of options and components for this installation. Click <strong>Install</strong> to start installation.</td>
</tr>
<tr>
<td>Oracle Net Configuration Assistant: Welcome</td>
<td>Click <strong>Cancel</strong>.</td>
</tr>
<tr>
<td>Oracle Net Configuration Assistant:</td>
<td>Click <strong>Yes</strong>.</td>
</tr>
<tr>
<td>Oracle Universal Installer: Configuration Tools</td>
<td>Click <strong>Exit</strong>.</td>
</tr>
<tr>
<td>Exit</td>
<td>The final screen of the Oracle Universal Installer is the End of Installation screen. Click <strong>Exit</strong> to exit the installer.</td>
</tr>
</tbody>
</table>
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:

\texttt{ORACLE_HOME/dg4msql/admin/initdg4msql.ora}

Where \texttt{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 \texttt{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 \texttt{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 \texttt{HS_FDS_CONNECT_INFO} is as follows:

\texttt{HS_FDS_CONNECT_INFO= host_name[/instance_name][/database_name]}

Where:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>\texttt{host_name}</td>
<td>is the host name or IP address of the machine hosting the SQL Server database.</td>
</tr>
<tr>
<td>\texttt{instance_name}</td>
<td>is the instance of SQL Server running on the machine.</td>
</tr>
<tr>
<td>\texttt{database_name}</td>
<td>is the SQL Server Database database name.</td>
</tr>
</tbody>
</table>

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

\texttt{HS_FDS_CONNECT_INFO= host_name//database_name}

\textbf{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. \textbf{Configure Oracle Net Listener for the Gateway}
2. \textbf{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:

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

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)
   
  
  }
}
Configure the Oracle Database for Gateway Access

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=dg4msq1)
  )
)
```

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/dg4msq1/admin directory where ORACLE_HOME is the directory under which the gateway is installed.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gateway_sid</td>
<td>specifies the SID of the gateway and matches the gateway SID specified in the tnsnames.ora file.</td>
</tr>
<tr>
<td>oracle_home_directory</td>
<td>specifies the Oracle home directory where the gateway resides.</td>
</tr>
<tr>
<td>dg4msq1</td>
<td>specifies the executable name of the Oracle Database Gateway for SQL Server.</td>
</tr>
</tbody>
</table>

See Also: Oracle Database Net Services Administrator’s Guide for information about changing the listener.ora file.
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>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect_descriptor</td>
<td>is the description of the object to connect to as specified when creating the database link, such as dg4msql.</td>
</tr>
<tr>
<td></td>
<td>Check the sqlnet.ora file in the Oracle database’s ORACLE_HOME for the following lines:</td>
</tr>
<tr>
<td></td>
<td>■ names.directory_path = (TNSNAMES, HOSTNAME)</td>
</tr>
<tr>
<td></td>
<td>■ names.default_domain = world</td>
</tr>
<tr>
<td></td>
<td>■ name.default_zone = world</td>
</tr>
<tr>
<td>TCP</td>
<td>is the TCP protocol used for TCP/IP connections.</td>
</tr>
<tr>
<td>host_name</td>
<td>specifies the machine where the gateway is running.</td>
</tr>
<tr>
<td>port_number</td>
<td>matches the port number used by the Oracle Net Listener that is listening for the gateway. The Oracle Net Listener’s port number can be found in the listener.ora file used by the Oracle Net Listener. See “Syntax of listener.ora File Entries” on page 9-3.</td>
</tr>
<tr>
<td>gateway_sid</td>
<td>specifies the SID of the gateway and matches the SID specified in the listener.ora file of the Oracle Net Listener that is listening for the gateway. See “Configure Oracle Net Listener for the Gateway” on page 9-3 for more information.</td>
</tr>
<tr>
<td>{HS=OK}</td>
<td>specifies that this connect descriptor connects to a non-Oracle system.</td>
</tr>
</tbody>
</table>

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

Where:

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

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

```sql
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.
Create SQL Server Views for Data Dictionary Support

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 `dg4msql`
- 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:
Configure the Gateway to Access Multiple SQL Server Databases

```
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 MSQL2 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 MSQL3 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@MSQL2;
SQL> SELECT * FROM ALL_USERS@MSQL3;
```
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"
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>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total disk space</td>
<td>4 GB</td>
<td>5 GB</td>
</tr>
</tbody>
</table>
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 (32-bit) operating systems:

- Microsoft Windows 2000 with Service Pack 1 or later. All editions, including Terminal Services and Microsoft Windows 2000 MultiLanguage Edition (MLE), are supported.
- Microsoft Windows Server 2003 - all editions
Step Through the Oracle Universal Installer

Installing Oracle Database Gateway for ODBC

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


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:

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

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Universal Installer: Welcome</td>
<td>Click Next.</td>
</tr>
</tbody>
</table>
| Oracle Universal Installer: Specify Home Details | Specify a name for the installation in the Name field. You can also choose not to edit the default setting of the Name field of the Specify Home Details screen.  
  The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to ORACLE_HOME. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen. |
  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. |
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 \texttt{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 \texttt{HS\_FDS\_CONNECT\_INFO} initialization parameter specifies the information required for connecting to the non-Oracle system. Set the \texttt{HS\_FDS\_CONNECT\_INFO} as follows:

\begin{verbatim}
HS\_FDS\_CONNECT\_INFO=dsn_value
\end{verbatim}

where \texttt{dsn_value} is the name of the system DSN defined in the Microsoft Windows ODBC Data Source Administrator.

\begin{itemize}
  \item \textbf{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.
\end{itemize}

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 \texttt{initsid.ora} file:

\begin{verbatim}
HS\_FDS\_CONNECT\_INFO=sqlserver7
\end{verbatim}

\texttt{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 \textbf{Start} menu, choose \textbf{Settings} > \textbf{Control Panel} and select \textbf{ODBC}.
2. Select the \textbf{System DSN} tab page to display the system data sources.
3. Click \textbf{Add}.
4. From the list of installed ODBC drivers, select the name of the driver that the data source will use. For example, select \textbf{SQL Server}.
5. Click \textbf{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.

\begin{itemize}
  \item \textbf{Note:} The name entered for the DSN must match the value of the initialization parameter \texttt{HS\_FDS\_CONNECT\_INFO} that is specified in \texttt{initsid.ora}.
\end{itemize}

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

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>host_name</code></td>
<td>is the name of the machine on which the gateway is installed.</td>
</tr>
<tr>
<td><code>port_number</code></td>
<td>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.</td>
</tr>
</tbody>
</table>

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:** If the ODBC driver supports Quoted Identifiers or Delimited Identifiers it should be turned on.
Configure Oracle Net for the Gateway

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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gateway_sid</td>
<td>specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the tnsnames.ora file.</td>
</tr>
<tr>
<td>oracle_home_directory</td>
<td>specifies the Oracle home directory where the gateway resides.</td>
</tr>
<tr>
<td>dg4odbc</td>
<td>specifies the executable name of the Oracle Database Gateway for ODBC.</td>
</tr>
</tbody>
</table>

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_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:

```sql
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**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect_descriptor</td>
<td>is the description of the object to connect to as specified when creating the database link, such as dg4odbc.</td>
</tr>
<tr>
<td>names.directory_path</td>
<td>parameter setting for the sqlnet.ora file: names.directory_path = (TNSNAMES)</td>
</tr>
<tr>
<td>Note:</td>
<td>The sqlnet.ora file is typically stored in ORACLE_HOME\network\admin.</td>
</tr>
<tr>
<td>TCP</td>
<td>is the TCP protocol used for TCP/IP connections.</td>
</tr>
<tr>
<td>host_name</td>
<td>specifies the machine where the gateway is running.</td>
</tr>
<tr>
<td>port_number</td>
<td>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 11-3.</td>
</tr>
<tr>
<td>gateway_sid</td>
<td>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 11-3 for more information.</td>
</tr>
<tr>
<td>(HS=OK)</td>
<td>specifies that this connect descriptor connects to a non-Oracle system.</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dblink</td>
<td>is the complete database link name.</td>
</tr>
<tr>
<td>tns_name_entry</td>
<td>specifies the Oracle Net connect descriptor specified in the tnsnames.ora file that identifies the gateway</td>
</tr>
</tbody>
</table>

After the database link is created you can verify the connection to the ODBC data source, as follows:

```
SQL> SELECT * FROM DUAL@dblink;
```
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 \texttt{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.

\textbf{See Also:} \textit{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 \texttt{dg4odbc}.
- The gateway is configured to access one ODBC data source named \texttt{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

\textbf{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:

- \texttt{dg4odbc2} for the gateway accessing data source dsn2.
- \texttt{dg4odbc3} for the gateway accessing data source dsn3.

\textbf{Create Two Initialization Parameter Files}
Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file: \texttt{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 \texttt{HS_FDS_CONNECT_INFO} parameter in the new files as follows:
For \texttt{initdg4odbc2.ora}, enter the following:
Configure the Gateway to Access Multiple ODBC Data Sources

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:

```plaintext
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
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
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
SQL> SELECT * FROM ALL_USERS@ODBC2;
SQL> SELECT * FROM ALL_USERS@ODBC3;
```
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"
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 12-3 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 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>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total disk space</td>
<td>4 GB</td>
<td>5 GB</td>
</tr>
</tbody>
</table>
Table 12–1  (Cont.) Hardware Requirements for Oracle Database Gateway for DRDA

<table>
<thead>
<tr>
<th>Requirement</th>
<th>For Microsoft Windows (32-bit)</th>
<th>For Microsoft Windows (64-bit)</th>
</tr>
</thead>
</table>
| Physical Memory     | 1 GB of real memory is the recommended minimum for running one instance of the gateway. Running additional instances of the Oracle Database Gateway for DRDA might require additional real memory or increased swap space to achieve reasonable performance. The total real memory requirement for each concurrent use of the gateway depends on the following factors:  
  - Number of concurrent TCP/IP connections opened by each user  
  - Number of data items being transferred between the gateway and the remote transaction program  
  - Additional factors such as configured network buffer size | Minimum of 1 GB                                               |
| Virtual memory      | Double the amount of RAM                                            | Double the amount of RAM                     |
| Video adapter       | 256 colors                                                          | 256 colors                                   |
| Processor           | 550 MHz minimum (On Microsoft Windows Vista, 800 MHz minimum)       | 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 (32-bit) operating systems:
Step Through the Oracle Universal Installer

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:


Step Through the Oracle Universal Installer

Table 12–2 describes the installation procedure for Oracle Database Gateway for DRDA.

<table>
<thead>
<tr>
<th>Screen</th>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Universal Installer: Welcome</td>
<td>Click Next.</td>
</tr>
<tr>
<td>Oracle Universal Installer: Specify Home Details</td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>The Path field in the Specify Home Details screen is where you specify the destination for your installation. You need not edit the path specification in the Path field. The default setting for this field points to ORACLE_HOME. After you set the fields in the Specify Home Details screen as necessary, click Next to continue. After loading the necessary information from the installation, the Oracle Universal Installer displays the Available Products screen.</td>
</tr>
<tr>
<td></td>
<td>b. Click Next.</td>
</tr>
<tr>
<td>Oracle Universal Installer: Summary</td>
<td>The Installation Summary screen enables you to review a tree list of options and components for this installation. Click Install to start installation.</td>
</tr>
<tr>
<td>Oracle Net Configuration Assistant: Welcome</td>
<td>Click Cancel.</td>
</tr>
</tbody>
</table>
Step Through the Oracle Universal 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.
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

### 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:

     ```sql
     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.

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

- **Schema privilege of CREATEIN, for example:**
  
  ```sql
  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:**
  
  ```sql
  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:

   ```sql
   SELECT CURRENT SERVER FROM any_table
   ```

   where `any_table` is a valid table with one or more rows.

   If the value returned by this query is blank or null, then the DRDA location name has not been established. Contact your system administrator to set a location name for the instance.
Configuring Oracle Database Gateway for DRDA

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

1. **Configure the Gateway Initialization Parameter File**
2. **Configure Oracle Net for the Gateway**
3. **Configure Two-Phase Commit**
4. **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 (e.g., "zos", "as400" and "luw") and version specific numbers (e.g., "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:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host_name</td>
<td>is the name of the machine on which the gateway is installed.</td>
</tr>
</tbody>
</table>
To direct the Oracle Net Listener to start the gateway in response to incoming connection requests, add an entry to the `listener.ora` file.

```
SID_LIST_LISTENER=
   {SID_LIST=
      {SID_DESC=
         {SID_NAME=gateway_sid}
         {ORACLE_HOME=oracle_home_directory}
         {PROGRAM=dg4db2}
      }
   }
```

Where:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>gateway_sid</td>
<td>specifies the SID of the gateway and matches the gateway SID specified in the connect descriptor entry in the <code>tnsnames.ora</code> file.</td>
</tr>
<tr>
<td>oracle_home_directory</td>
<td>specifies the Oracle home directory where the gateway resides.</td>
</tr>
<tr>
<td>dg4db2</td>
<td>specifies the executable name of the Oracle Database Gateway for DRDA.</td>
</tr>
</tbody>
</table>

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_NAME=.
         }
      }
      {SID_DESC=.
         {SID_NAME=.
         }
      }
      {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.

```
connectDescriptor=
  (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

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>connect_descriptor</td>
<td>is the description of the object to connect to as specified when creating the database link, such as dg4db2.</td>
</tr>
<tr>
<td>check sqlnet.ora file for the following parameter setting: names.directory_path = (TNSNAMES)</td>
<td></td>
</tr>
<tr>
<td>Note: The sqlnet.ora file is typically stored in ORACLE_HOME\network\admin.</td>
<td></td>
</tr>
<tr>
<td>TCP</td>
<td>is the TCP protocol used  for TCP/IP connections.</td>
</tr>
<tr>
<td>host_name</td>
<td>specifies the machine where the gateway is running.</td>
</tr>
<tr>
<td>port_number</td>
<td>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 14-2.</td>
</tr>
<tr>
<td>gateway_sid</td>
<td>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 &quot;Configure Oracle Net Listener for the Gateway&quot; on page 14-2 for more information.</td>
</tr>
<tr>
<td>(HS=OK)</td>
<td>specifies that this connect descriptor connects to a non-Oracle system.</td>
</tr>
</tbody>
</table>

Configuring tnsnames.ora for Multiple Listeners

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

```
connectDescriptor=
  (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)
    )
  )
```

TCP is the TCP protocol used for TCP/IP connections.

host_name specifies the machine where the gateway is running.

port_number 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 14-2.

gateway_sid 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 14-2 for more information.

(HS=OK) specifies that this connect descriptor connects to a non-Oracle system.
(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
SQL> CREATE PUBLIC DATABASE LINK dblink CONNECT TO 2 "user" IDENTIFIED BY "password" USING 'tns_name_entry';
```

Where:

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

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

```sql
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.
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:

- \texttt{dg4db22} for the gateway accessing database \texttt{db2}.
- \texttt{dg4db23} for the gateway accessing database \texttt{db3}.

Create Two Initialization Parameter Files
Create an initialization parameter file for each instance of the gateway by copying the original initialization parameter file:

\begin{verbatim}
> cd \texttt{ORACLE_HOME\dg4db2\admin}
> copy initdg4db2.ora initdg4db22.ora
> copy initdg4db2.ora initdg4db23.ora
\end{verbatim}

Add Entries to listener.ora
Add two new entries to the Oracle Net Listener configuration file, \texttt{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:

\begin{verbatim}
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)
 )
)
\end{verbatim}
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:

```plaintext
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
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;
```
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.
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 **GRANT**s 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.
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/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.
Obsoleted Parameters

The following parameters are obsolete for the 11g version. Remove them from your configuration files:

- MODE
- SERVER_PATH
- DRDA_OVERRIDE_FROM_CODEPAGE
- DRDA_OVERRIDE_TO_CODEPAGE
- ERROR_LOGGING
- ERROR_REPORTING
- ERRORTAG
- GATEWAY_SID
- GROUP_BY_OFF
- GTWDEBUG
- INCREMENT_CURSORS
- DRDA_CALLDESC_STMT
- DRDA_CALLDESC_PROC

Bind Gateway Package

When upgrading to 11g release you must rebind the gateway package if you have changed any of the following initialization parameters:

- DRDA_DISABLE_CALL
- DRDA_ISOLATION_LEVEL
- DRDA_PACKAGE_COLLID
- DRDA_PACKAGE_CONSTOKEN
- DRDA_PACKAGE_NAME
- DRDA_PACKAGE_OWNER
- DRDA_PACKAGE_SECTIONS

Install/Upgrade Data Dictionary Views

You must install or upgrade the data dictionary views when upgrading the Oracle Database Gateway for DRDA. Refer to Chapter 14, "Configuring Oracle Database Gateway for DRDA" for more information on creating data dictionary views.
Part VIII, "Removing Oracle Database Gateway" 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 an 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.tmpl`.

  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" 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 (32-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.
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**.


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
   ```

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

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://www.oracle.com/technetwork/documentation/index.html](http://www.oracle.com/technetwork/documentation/index.html)

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:

<table>
<thead>
<tr>
<th>Status</th>
<th>Result Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration assistant succeeded</td>
<td>0</td>
</tr>
<tr>
<td>Configuration assistant failed</td>
<td>1</td>
</tr>
<tr>
<td>Configuration assistant cancelled</td>
<td>-1</td>
</tr>
</tbody>
</table>

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.
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**
Oracle Database Gateway for Informix Initialization Parameters

This section lists all the initialization file parameters that can be set for the Oracle Database Gateway for Informix. They are as follows:

- HS_DB_DOMAIN
- HS_DB_INTERNAL_NAME
- HS_DB_NAME
- HS_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TIME_ZONE
- HS_TRANSACTION_MODEL
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER
- HS_FDS_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 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_DESCRIBE_CACHE_HWM
- HS_LANGUAGE
- HS_LONG_PIECE_TRANSFER_SIZE
- HS_OPEN_CURSORS
- HS_RPC_FETCH_REBLOCKING
- HS_RPC_FETCH_SIZE
- HS_TIME_ZONE
- HS_TRANSACTION_MODEL
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER
- HS_FDS_RECOVERY_ACCOUNT
- HS_FDS_RECOVERY_PWD
- HS_FDS_TRACE_LEVEL
- HS_FDS_TRANSACTION_LOG
- HS_FDS_FETCH_ROWS
- 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
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_DEFAULT_OWNER**
- **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 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_DEFAULT_OWNER**
- **HS_FDS_RECOVERY_ACCOUNT**
- **HS_FDS_RECOVERY_PWD**
- **HS_FDS_REPORT_REAL_AS_DOUBLE**
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_TIME_ZONE
- IFILE
- HS_FDS_CONNECT_INFO
- HS_FDS_DEFAULT_OWNER
- 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_DEFAULT_OWNER
■ 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_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_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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>None</td>
</tr>
<tr>
<td>Range of values</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>WORLD</td>
</tr>
<tr>
<td>Range of values</td>
<td>1 to 199 characters</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>01010101</td>
</tr>
<tr>
<td>Range of values</td>
<td>1 to 16 hexadecimal characters</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>HO</td>
</tr>
<tr>
<td>Range of values</td>
<td>1 to 8 characters</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>100</td>
</tr>
<tr>
<td>Range of values</td>
<td>1 to 4000</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>System-specific</td>
</tr>
<tr>
<td>Range of values</td>
<td>Any valid language name (up to 255 characters)</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>64 KB</td>
</tr>
<tr>
<td>Range of values</td>
<td>Any value up to 2 GB</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>50</td>
</tr>
<tr>
<td>Range of values</td>
<td>1 to the value of OPEN_CURSORS initialization parameter of Oracle database</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>ON</td>
</tr>
<tr>
<td>Range of values</td>
<td>OFF or ON</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>50000</td>
</tr>
<tr>
<td>Range of values</td>
<td>1 to 100000000</td>
</tr>
</tbody>
</table>

Tunes internal data buffering to optimize the data transfer rate between the server and the agent process.

Increasing the value can reduce the number of network round-trips needed to transfer a given amount of data, but also tends to increase data bandwidth and to reduce latency as measured between issuing a query and completion of all fetches for the query. Nevertheless, increasing the fetch size can increase latency for the initial fetch results of a query, because the first fetch results are not transmitted until additional data is available.
**HS_TIME_ZONE**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value for '[+</td>
<td>-]hh:mm'</td>
</tr>
<tr>
<td>Range of values for '[+</td>
<td>-]hh:mm'</td>
</tr>
</tbody>
</table>

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—which formerly Greenwich Mean Time). For example:

\[
\text{HS\_TIME\_ZONE} = [+ | -] \text{hh:mm}
\]

**HS_TRANSACTION_MODEL**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>COMMIT_CONFIRM</td>
</tr>
<tr>
<td>Range of Values</td>
<td>COMMIT_CONFIRM, READ_ONLY, READ_ONLY_AUTOCOMMIT, SINGLE_SITE, SINGLE_SITE_AUTOCOMMIT</td>
</tr>
</tbody>
</table>

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.

- 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.
**IFILE**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default value</td>
<td>None</td>
</tr>
<tr>
<td>Range of values</td>
<td>Valid parameter file names</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>None</td>
</tr>
<tr>
<td>Range of Values</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

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

\begin{verbatim}
HS_FDS_CONNECT_INFO= host\_name//database\_name
\end{verbatim}

**For Oracle Database Gateway for ODBC:**

\begin{verbatim}
HS_FDS_CONNECT_INFO=dsn_value
\end{verbatim}

where \texttt{dsn\_value} is the name of the system DSN defined in the Microsoft Windows ODBC Data Source Administrator.

**For Oracle Database Gateway for DRDA:**

\begin{verbatim}
HS_FDS_CONNECT_INFO=IP\_address:Port\_number\_Database\_name,Type
\end{verbatim}

Where \texttt{IP\_address} is the hostname or ip address of the DB2 DRDA server, 

\texttt{Port\_number} is the port number of the DB2 DRDA server.

\texttt{Database\_name} is the database name of the DB2 server.

\texttt{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,

\begin{verbatim}
HS_FDS_CONNECT_INFO=[2001:0db8:20C:F1FF:FEC6:38AF]:1300/DB2M,ZOS
\end{verbatim}

**HS_FDS_DEFAULT_OWNER**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>None</td>
</tr>
<tr>
<td>Range of Values</td>
<td>Not applicable</td>
</tr>
</tbody>
</table>

The name of the table owner that is used for the non-Oracle database tables if an owner is not specified in the SQL statements.

**Note:** If this parameter is not specified and the owner is not explicitly specified in the SQL statement, then the user name of the Oracle user or the user name specified when creating the database link is used.
**HS_FDS_PROC_IS_FUNC**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>FALSE</td>
</tr>
<tr>
<td>Range of Values</td>
<td>TRUE, FALSE</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>RECOVER</td>
</tr>
<tr>
<td>Range of values</td>
<td>Any valid user ID</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>none</td>
</tr>
<tr>
<td>Range of values</td>
<td>Any valid password</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>FALSE</td>
</tr>
<tr>
<td>Range of Values</td>
<td>TRUE, FALSE</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>OFF</td>
</tr>
<tr>
<td>Range of Values</td>
<td>OFF, ON, DEBUG</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>HS_TRANSACTION_LOG</td>
</tr>
<tr>
<td>Range of Values</td>
<td>Any valid table name</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>FALSE</td>
</tr>
<tr>
<td>Range of Values</td>
<td>TRUE, FALSE</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>100</td>
</tr>
<tr>
<td>Range of Values</td>
<td>Any integer between 1 and 1000</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_FETCH_ROWS=num</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>None</td>
</tr>
<tr>
<td>Range of Values</td>
<td>Refer to Chapter 4, 'Developing Applications' in Oracle Database Gateway for DRDA User’s Guide</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_CAPABILITY= {FUNCTION/{ON</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>CHG for DB2 UDB for iSeries, CS for DB2 UDB for z/OS, DB2/UDB</td>
</tr>
<tr>
<td>Range of Values</td>
<td>{CHG</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_ISOLATION_LEVEL=(CHG</td>
</tr>
</tbody>
</table>

If the HS_FDS_ISOLATION_LEVEL is set to CHG then the data will be converted to the appropriate character set.

If the HS_FDS_ISOLATION_LEVEL is set to CS then the data will be converted to the appropriate character set and the data will be converted to the appropriate numeric type.

If the HS_FDS_ISOLATION_LEVEL is set to RR then the data will be converted to the appropriate numeric type and the data will be converted to the appropriate character set.

If the HS_FDS_ISOLATION_LEVEL is set to ALL then the data will be converted to the appropriate numeric type, the data will be converted to the appropriate character set, and the data will be converted to the appropriate numeric type.

If the HS_FDS_ISOLATION_LEVEL is set to NC then the data will be converted to the appropriate numeric type, the data will be converted to the appropriate character set, and the data will be converted to the appropriate numeric type.
**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>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHG</td>
<td>Change (default for DB2 UDB for iSeries)</td>
</tr>
<tr>
<td>CS</td>
<td>Cursor Stability (default for DB2 UDB for Linux, UNIX, and Windows, and DB2 UDB for z/OS)</td>
</tr>
<tr>
<td>RR</td>
<td>Repeatable Read</td>
</tr>
<tr>
<td>ALL</td>
<td>ALL</td>
</tr>
<tr>
<td>NC</td>
<td>No Commit</td>
</tr>
</tbody>
</table>

**HS_FDS_PACKAGE_COLLID**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>ORACLEGTW</td>
</tr>
<tr>
<td>Range of Values</td>
<td>An alphanumeric string 1 to 18 characters in length</td>
</tr>
<tr>
<td>Syntax</td>
<td><strong>HS_FDS_PACKAGE_COLLID=collection_id</strong></td>
</tr>
</tbody>
</table>

**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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>0 (no timeout)</td>
</tr>
<tr>
<td>Range of Values</td>
<td>0.0-9999 (minutes)</td>
</tr>
<tr>
<td>Syntax</td>
<td><strong>HS_IDLE_TIMEOUT=num</strong></td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>FALSE</td>
</tr>
<tr>
<td>Range of Values</td>
<td>FALSE</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_MBCS_TO_GRAPHIC=(FALSE</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>FALSE</td>
</tr>
<tr>
<td>Range of Values</td>
<td>FALSE</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_GRAPHIC_TO_MBCS=(FALSE</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>CHAR</td>
</tr>
<tr>
<td>Range of Values</td>
<td>CHAR</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_TIMESTAMP_MAPPING={(CHAR</td>
</tr>
</tbody>
</table>

If set to CHAR (default), DB2 timestamp would be mapped to CHAR(26). DB2 timestamp would be mapped to Oracle timestamp if set to TIMESTAMP.

**HS_FDS_QUOTE_IDENTIFIER**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>TRUE for Sybase</td>
</tr>
<tr>
<td>Range of Values</td>
<td>TRUE</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_QUOTE_IDENTIFIER={(FALSE</td>
</tr>
</tbody>
</table>
By default, the gateway will quote identifiers if the FDS supports it. However, we give the user the ability to overwrite the behavior. \texttt{HS\_FDS\_QUOTE\_IDENTIFIER} overrides the target’s ability to support quote identifier depending on the value provided.

**HS\_NLS\_LENGTH\_SEMANTICS**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>BYTE</td>
</tr>
<tr>
<td>Range of values</td>
<td>BYTE | CHAR</td>
</tr>
<tr>
<td>Syntax</td>
<td>$\texttt{HS_NLS_LENGTH_SEMANTICS} = { \texttt{BYTE} | \texttt{CHAR} }$</td>
</tr>
</tbody>
</table>

This release of gateway has Character Semantics functionality equivalent to the Oracle database Character Semantics, that is, \texttt{NLS\_LENGTH\_SEMANTICS}. When \texttt{HS\_NLS\_LENGTH\_SEMANTICS} is set to \texttt{CHAR}, the (VAR)CHAR columns of non-Oracle database are to be interpreted as having \texttt{CHAR} semantics. The only situation the gateway does not honor the \texttt{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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>OFF</td>
</tr>
<tr>
<td>Range of Values</td>
<td>OFF | LOCAL | REMOTE | ALL</td>
</tr>
<tr>
<td>Syntax</td>
<td>$\texttt{HS_KEEP_REMOTE_COLUMN_SIZE} = { \texttt{OFF} | \texttt{LOCAL} | \texttt{REMOTE} | \texttt{ALL} }$</td>
</tr>
<tr>
<td>Parameter type</td>
<td>String</td>
</tr>
</tbody>
</table>

\texttt{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 \texttt{REMOTE}, the expansion is suppressed between the non-Oracle database to the gateway. When it is set to \texttt{LOCAL}, the expansion is suppressed between the gateway and Oracle database. When it is set to \texttt{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 \texttt{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 \texttt{INSERT}, \texttt{UPDATE}, or \texttt{DELETE}.

**HS\_FDS\_REMOTE\_DB\_CHARSET**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>None</td>
</tr>
<tr>
<td>Range of values</td>
<td>Not Applicable</td>
</tr>
</tbody>
</table>
Initialization Parameter Description

This parameter is valid only when $\text{HS\_LANGUAGE}$ 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 $\text{HS\_LANGUAGE}=\text{AL32UTF8}$. 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 $\text{HS\_FDS\_REMOTE\_DB\_CHARSET}$ is set to the corresponding non-Oracle database character set. For example, $\text{HS\_FDS\_REMOTE\_DB\_CHARSET}=\text{KO16KSC5601}$.

**HS\_FDS\_SUPPORT\_STATISTICS**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>TRUE</td>
</tr>
<tr>
<td>Range of values</td>
<td>(TRUE</td>
</tr>
<tr>
<td>Syntax</td>
<td>$\text{HS_FDS_SUPPORT_STATISTICS= (TRUE</td>
</tr>
</tbody>
</table>

We gather statistics from the non-Oracle database by default. You can choose to disable the gathering of remote database statistics by setting the $\text{HS\_FDS\_SUPPORT\_STATISTICS}$ parameter to FALSE.

**HS\_FDS\_RSET\_RETURN\_ROWCOUNT**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>FALSE</td>
</tr>
<tr>
<td>Range of values</td>
<td>(TRUE</td>
</tr>
<tr>
<td>Syntax</td>
<td>$\text{HS_FDS_RSET_RETURN_ROWCOUNT= (TRUE</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>64</td>
</tr>
<tr>
<td>Range of values</td>
<td>(64</td>
</tr>
<tr>
<td>Syntax</td>
<td>$\text{HS_FDS_SQLLEN_INTERPRETATION= (64</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>CLEARTEXT</td>
</tr>
<tr>
<td>Range of values</td>
<td>(CLEARTEXT</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_AUTHENTICATE_METHOD= (CLEARTEXT</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>NONE</td>
</tr>
<tr>
<td>Range of values</td>
<td>NONE</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_ENCRYPT_SESSION = NONE</td>
</tr>
</tbody>
</table>

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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>none</td>
</tr>
<tr>
<td>Range of values</td>
<td>path to truststore file</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_TRUSTSTORE_FILE = path to truststore file</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Value</td>
<td>none</td>
</tr>
<tr>
<td>Range of values</td>
<td>password</td>
</tr>
<tr>
<td>Syntax</td>
<td>HS_FDS_TRUSTSTORE_PASSWORD= password</td>
</tr>
</tbody>
</table>

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>
<thead>
<tr>
<th>Reason</th>
<th>Name of Parameter Needed</th>
<th>Your Specific Parameters Here</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle home of the gateway</td>
<td>▪ ORACLE_HOME</td>
<td></td>
</tr>
<tr>
<td>System ID of the gateway</td>
<td>▪ ORACLE_SID</td>
<td></td>
</tr>
<tr>
<td>Configuring TCP/IP</td>
<td>▪ Local Host name, Domain Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▪ IP Address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▪ Network Mask</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▪ Name Server IP Address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▪ DRDA server Host name or IP Address</td>
<td></td>
</tr>
<tr>
<td></td>
<td>▪ DRDA server Service Port Number</td>
<td></td>
</tr>
<tr>
<td>Recovery user ID</td>
<td>▪ HS_FDS_RECOVERY_ACCOUNT</td>
<td></td>
</tr>
<tr>
<td>Recovery Password</td>
<td>▪ HS_FDS_RECOVERY_PWD</td>
<td></td>
</tr>
<tr>
<td>Remote Database Name</td>
<td>▪ DRDA_REMOTE_DB_NAME</td>
<td></td>
</tr>
<tr>
<td>Connection Parameter</td>
<td>▪ DRDA_CONNECT_PARM</td>
<td></td>
</tr>
<tr>
<td>Remote collection ID</td>
<td>▪ HS_FDS_PACKAGE_COLLID</td>
<td></td>
</tr>
<tr>
<td>Remote package name</td>
<td>▪ DRDA_PACKAGE_NAME</td>
<td></td>
</tr>
<tr>
<td>Owner ID of DRDA package</td>
<td>▪ DRDA_PACKAGE_OWNER</td>
<td></td>
</tr>
<tr>
<td>DB Name used with Oracle database</td>
<td>▪ HS_DB_NAME</td>
<td></td>
</tr>
<tr>
<td>DB Domain used with Oracle database</td>
<td>▪ HS_DB_DOMAIN</td>
<td></td>
</tr>
</tbody>
</table>
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 iSeries, 13-3
  DB2 UDB for Linux, UNIX, and Windows, 13-4
  DB2/OS390, 13-1
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

collection privilege - CREATE IN
  configuration worksheet, D-2
  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
copy privilege
  configuration worksheet, D-2
  DB2/OS390, 13-1
create in privilege
  configuration worksheet, D-2
  DB2/OS390, 13-2
create in privilege, DB2 UDB for Linux, UNIX, and Windows, 13-4
create tab privilege
  DB2 UDB for Linux, UNIX, and Windows, 13-4
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
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 Windows 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

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

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-11
tuning LONG data transfer, C-10
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-13
HS_FDS_CONNECT_INFO initialization parameter specifying connection information, 11-2
HS_FDS_DEFAULT_OWNER initialization parameter, C-14
HS_FDS_FETCH_ROWS parameter, C-17
HS_FDS_ISOLATION_LEVEL parameter, C-17
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
<table>
<thead>
<tr>
<th><strong>DB2/OS390</strong></th>
<th>13-2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>HS_FDS_RECOVERY_PWD initialization parameter</strong>, C-17</td>
<td></td>
</tr>
<tr>
<td><strong>HS_FDS_RECOVERY_PWD parameter</strong></td>
<td></td>
</tr>
<tr>
<td>DB2 UDB for Linux, UNIX, and Windows, 13-4</td>
<td></td>
</tr>
<tr>
<td>DB2/OS390, 13-2, 13-3</td>
<td></td>
</tr>
<tr>
<td><strong>HS_FDS_RESULTSET_SUPPORT initialization parameter</strong>, C-16</td>
<td></td>
</tr>
<tr>
<td><strong>HS_FDS_TRACELEVEL initialization parameter</strong>, C-16</td>
<td></td>
</tr>
<tr>
<td>enabling agent tracing, C-2</td>
<td></td>
</tr>
<tr>
<td><strong>HS_FDS_TRANSACTION_LOG initialization parameter</strong>, C-16</td>
<td></td>
</tr>
<tr>
<td><strong>HS_KEEP_REMOTE_COLUMN_SIZE initialization parameter</strong>, C-20</td>
<td></td>
</tr>
<tr>
<td><strong>HS_LANGUAGE initialization parameter</strong>, C-9</td>
<td></td>
</tr>
<tr>
<td><strong>HS_LONG_PIECE_TRANSFER_SIZE initialization parameter</strong>, C-10</td>
<td></td>
</tr>
<tr>
<td><strong>HS_OPEN_CURSORS initialization parameter</strong>, C-11</td>
<td></td>
</tr>
<tr>
<td><strong>HS_RPC_FETCH_REBLOCKING initialization parameter</strong>, C-11</td>
<td></td>
</tr>
<tr>
<td><strong>HS_RPC_FETCH_SIZE initialization parameter</strong>, C-11</td>
<td></td>
</tr>
<tr>
<td><strong>HS_TIME_ZONE initialization parameter</strong>, C-12</td>
<td></td>
</tr>
<tr>
<td><strong>HS_TRANSACTION_LOG</strong></td>
<td>3-7, 5-7, 7-7, 9-8</td>
</tr>
<tr>
<td><strong>HS_TRANSACTION_LOG table</strong></td>
<td></td>
</tr>
<tr>
<td><strong>DB2 UDB for iSeries</strong>, 13-5</td>
<td></td>
</tr>
<tr>
<td><strong>DB2 UDB for Linux, UNIX, and Windows</strong>, 13-4</td>
<td></td>
</tr>
<tr>
<td><strong>IFILE initialization parameter</strong>, C-13</td>
<td></td>
</tr>
<tr>
<td>inbound connections</td>
<td></td>
</tr>
<tr>
<td>processing, 15-2</td>
<td></td>
</tr>
<tr>
<td><strong>Informix Server, Oracle Database Gateway</strong>, 4-1</td>
<td></td>
</tr>
<tr>
<td><strong>Informix, Oracle Database Gateway</strong>, 4-1</td>
<td></td>
</tr>
<tr>
<td><strong>initialization parameter file</strong></td>
<td></td>
</tr>
<tr>
<td>customizing, 3-2, 5-1, 7-1, 9-2, C-1</td>
<td></td>
</tr>
<tr>
<td><strong>initialization parameters</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Heterogeneous Services (HS)</strong>, 11-1</td>
<td></td>
</tr>
<tr>
<td><strong>Heterogeneous Services (HS) initialization parameters</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Oracle Database Gateway for ODBC</strong>, 11-1</td>
<td></td>
</tr>
<tr>
<td>initSID.ora file, 3-2, 5-2, 7-2, 9-2</td>
<td></td>
</tr>
<tr>
<td>installActions.log file, B-2</td>
<td></td>
</tr>
<tr>
<td><strong>installation</strong></td>
<td></td>
</tr>
<tr>
<td>accessing installation software, 1-3</td>
<td></td>
</tr>
<tr>
<td><strong>checklists</strong></td>
<td></td>
</tr>
<tr>
<td><strong>DRDA server</strong>, 13-1</td>
<td></td>
</tr>
<tr>
<td><strong>downloading software from Oracle Technology Network</strong>, 1-3</td>
<td></td>
</tr>
<tr>
<td><strong>errors</strong></td>
<td></td>
</tr>
<tr>
<td><strong>log session</strong>, B-2</td>
<td></td>
</tr>
<tr>
<td>while configuration assistant runs, B-2</td>
<td></td>
</tr>
<tr>
<td><strong>log files</strong>, B-2</td>
<td></td>
</tr>
<tr>
<td><strong>noninteractive error handling</strong>, B-3</td>
<td></td>
</tr>
<tr>
<td><strong>procedure</strong>, 1-4</td>
<td></td>
</tr>
<tr>
<td><strong>response files</strong>, B-3</td>
<td></td>
</tr>
<tr>
<td>reviewing a log of an installation session, B-2</td>
<td></td>
</tr>
<tr>
<td><strong>troubleshooting</strong>, B-1, B-3</td>
<td></td>
</tr>
<tr>
<td><strong>installation software, accessing</strong>, 1-3</td>
<td></td>
</tr>
<tr>
<td><strong>installations</strong></td>
<td></td>
</tr>
<tr>
<td><strong>log file</strong>, B-2</td>
<td></td>
</tr>
<tr>
<td><strong>isolation level</strong>, HS_FDS_ISOLATION_LEVEL, C-18</td>
<td></td>
</tr>
</tbody>
</table>
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-17
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 UDB for Linux, UNIX, and Windows, 13-4
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
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
evironment 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