

Oracle® Database

Installation Guide

10g Release 2 (10.2) for Apple Mac OS X (Intel)

B25286-01

March 2009

Oracle Database Installation Guide, 10g Release 2 (10.2) for Apple Mac OS X (Intel)

B25286-01

Copyright © 2009, Oracle and/or its affiliates. All rights reserved.

Primary Author: Brintha Bennet

Contributing Authors: Kevin Flood, Pat Huey, Clara Jaeckel, Emily Murphy, Terri Winters, Lyju Vadassery, Ashmita Bose

Contributors: David Austin, Subhranshu Banerjee, Mark Bauer, Robert Chang, Jonathan Creighton, Sudip Datta, Padmanabhan Ganapathy, Thirumaleshwara Hasandka, Joel Kallman, George Kotsovolos, Simon Law, Richard Long, Rolly Lv, Padmanabhan Manavazhi, Sreejith Minnanghat, Krishna Mohan, Rajendra Pingte, Hanlin Qian, Janelle Simmons, Preeti Shukla, Roy Swonger, Douglas Williams

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

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

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

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

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

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

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

Contents

Preface	ix
Audience	ix
Documentation Accessibility	ix
Command Syntax	x
Accessing Documentation	x
Related Documentation	xi
Typographic Conventions	xi
1 Installation Overview	
Overview of Oracle Database Installation	1-1
Oracle Database Installation Methods	1-2
Interactive Installation Methods	1-2
Automated Installation Methods Using Response Files	1-3
Oracle Database Installation Types	1-3
Database Configuration Options	1-4
Preconfigured Database Types	1-4
Installation Choices that Affect Database Creation	1-4
Creating a Database After Installation	1-4
Available Installation Options	1-5
Database Storage Options	1-5
File System	1-5
Installation Considerations	1-5
Hardware and Software Certification	1-5
Multiple Oracle Homes	1-6
Installing the Software on a System with an Existing Oracle Installation	1-6
2 Preinstallation Tasks	
Logging In to the System as root	2-1
Checking the Hardware Requirements	2-2
Checking the Software Requirements	2-3
Checking the Network Setup	2-3
Configuring Name Resolution	2-4
Installing on DHCP Computers	2-4
Installing on Multihomed Computers	2-4
Installing on Computers with Multiple Aliases	2-5

Installing on Non-Networked Computers	2-5
Creating Required Operating System Groups and Users	2-6
Creating the Oracle Inventory Group	2-7
Creating the OSDBA Group	2-8
Creating an OSOPER Group (Optional)	2-9
Creating the Oracle Software Owner User	2-9
Determining Whether an Oracle Software Owner User Exists	2-9
Creating an Oracle Software Owner User	2-10
Modifying an Oracle Software Owner User	2-10
Configuring Kernel Parameters.....	2-11
Identifying Required Software Directories	2-13
Oracle Base Directory	2-13
Oracle Inventory Directory	2-14
Oracle Home Directory	2-14
Identifying or Creating an Oracle Base Directory	2-15
Identifying an Existing Oracle Base Directory	2-15
Creating an Oracle Base Directory.....	2-16
Choosing a Storage Option for Oracle Database and Recovery Files.....	2-17
Creating Directories for Oracle Database or Recovery Files	2-17
Guidelines for Placing Oracle Database Files on a File System.....	2-18
Guidelines for Placing Oracle Recovery Files on a File System	2-18
Creating Required Directories.....	2-19
Stopping Existing Oracle Processes.....	2-20
Configuring the Oracle User’s Environment	2-21

3 Installation Tasks

Preinstallation Considerations	3-1
Performing Multiple Oracle Database Installations in Noninteractive Mode	3-1
Reviewing Component-Specific Installation Guidelines	3-1
Accessing the Installation Software	3-2
Downloading and Extracting Oracle Software from the OTN Web Site	3-2
Downloading the Installation Archive Files	3-2
Extracting the Installation Files.....	3-3
Copy the Oracle Database Software to a Hard Disk.....	3-3
Installing the Oracle Database Software	3-4
Reviewing Product-Specific Installation Guidelines	3-4
Running the Oracle Universal Installer	3-4

4 Postinstallation Tasks

Required Postinstallation Tasks.....	4-1
Downloading and Installing Patches	4-1
Configuring Oracle Products.....	4-2
Recommended Postinstallation Tasks.....	4-2
Creating a Backup of the root.sh Script	4-2
Configuring New or Upgraded Databases.....	4-2
Setting Up User Accounts	4-3
Setting the NLS_LANG Environment Variable.....	4-3

Generating the Client Static Library	4-3
Required Product-Specific Postinstallation Tasks	4-4
Configuring Oracle Net Services	4-4
Configuring Oracle Label Security	4-5
Installing Natively Compiled Java Libraries for Oracle JVM and Oracle <i>interMedia</i>	4-5
Installing Oracle Text Supplied Knowledge Bases	4-5
Configuring Oracle Messaging Gateway	4-5
Modifying the listener.ora File for External Procedures	4-6
Modifying the tnsnames.ora File for External Procedures	4-7
Setting Up the mgw.ora Initialization File	4-7
Configuring Oracle Precompilers	4-8
Configuring Pro*C/C++	4-8
Configuring OCCI	4-8
Installing Oracle Database 10g Products from the Companion CD	4-8
5 Getting Started with Oracle Database	
Checking the Installed Oracle Database Contents and Directory Location	5-1
Reviewing Accounts and Passwords	5-1
Unlocking and Resetting User Passwords	5-3
Using SQL*Plus to Unlock Accounts and Reset Passwords	5-4
Identifying Databases	5-4
Locating the Server Parameter File	5-5
Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files	5-5
Identifying Tablespaces and Data Files	5-5
Locating Redo Log Files	5-6
Locating Control Files	5-6
6 Removing Oracle Software	
Overview	6-1
Identifying All Instances	6-1
Removing an Oracle Database	6-1
Removing Oracle Software	6-2
A Installing and Configuring Oracle Database Using Response Files	
How Response Files Work	A-1
Reasons for Using Silent Mode or Noninteractive Mode	A-2
General Procedure for Using Response Files	A-2
Creating the oraInst.loc File	A-3
Preparing a Response File	A-3
Editing a Response File Template	A-3
Recording a Response File	A-5
Running Oracle Universal Installer Using a Response File	A-6
Running NetCA Using a Response File	A-7
Running Database Configuration Assistant Using a Response File	A-8
Using Database Configuration Assistant in Noninteractive Mode	A-8
Using Database Configuration Assistant in Silent Mode	A-8

Running Database Configuration Assistant in Noninteractive or Silent Mode.....	A-8
--	-----

B Cloning an Oracle Home

C Optimal Flexible Architecture

Overview of the Optimal Flexible Architecture Standard	C-1
Characteristics of an Optimal Flexible Architecture Compliant Installation	C-1
Changes to the Optimal Flexible Architecture for Oracle Database 10g.....	C-3
Implementing Optimal Flexible Architecture	C-3
File Systems.....	C-3
Number of File Systems	C-4
Naming Conventions	C-4
Naming Mount Points for Very Large Databases (VLDBs).....	C-4
Naming Directories.....	C-4
Oracle Base Directory Naming Convention.....	C-4
Referring to Path Names.....	C-5
Oracle Home Directory Naming Convention.....	C-5
Naming Subdirectories	C-5
Naming Database Files.....	C-6
Separating Segments with Different Requirements.....	C-6
Naming Tablespaces.....	C-7
Exploiting the Optimal Flexible Architecture Structure for Oracle Files.....	C-7
Optimal Flexible Architecture File Mapping	C-8
Improving Reliability and Performance.....	C-9
Disk Mirroring.....	C-9
Disk Striping	C-9

D Managing Oracle Database Port Numbers

About Managing Ports	D-1
Viewing Port Numbers and Access URLs	D-2
Port Numbers and Protocols of Oracle Components	D-2
Changing the <i>i</i> SQL*Plus Ports	D-3
Changing the Oracle Ultra Search Ports	D-4
Changing the Oracle XML DB Ports.....	D-4

E Configuring Oracle Database Globalization Support

About NLS_LANG Environment Variable.....	E-1
Running Oracle Universal Installer in Different Languages	E-2

F Troubleshooting

Verify Requirements.....	F-1
What to Do If an Installation Error Occurs?.....	F-1
Reviewing the Log of an Installation Session	F-2
Troubleshooting Configuration Assistants	F-2
Configuration Assistant Failure.....	F-3
Fatal Errors.....	F-3

Silent-Mode Response File Error Handling.....	F-3
Cleaning Up After a Failed Installation.....	F-4

G Frequently Asked Questions About Installation

Installing Oracle Database or Oracle Database Client	G-1
Installing Oracle Database Tools	G-3
Installing Oracle Database with Oracle Applications	G-5

Index

Preface

This guide describes how to install and configure Oracle Database 10g release 2 (10.2) on Apple Mac OS X (Intel).

Audience

This guide is intended for anyone responsible for installing Oracle Database 10g release 2 (10.2) on a single Apple Mac OS X (Intel) system.

Documentation Accessibility

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

To reach AT&T Customer Assistants, dial 711 or 1.800.855.2880. An AT&T Customer Assistant will relay information between the customer and Oracle Support Services at 1.800.223.1711. Complete instructions for using the AT&T relay services are available at <http://www.consumer.att.com/relay/tty/standard2.html>. After the AT&T Customer Assistant contacts Oracle Support Services, an Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process.

Command Syntax

UNIX command syntax appears in monospace font. The dollar character (\$), number sign (#), or percent character (%) are UNIX command prompts. Do not enter them as part of the command. The following command syntax conventions are used in this guide:

Convention	Description
backslash \	A backslash is the UNIX command continuation character. It is used in command examples that are too long to fit on a single line. Enter the command as displayed (with a backslash) or enter it on a single line without a backslash: <pre>dd if=/dev/rdskc0t1d0s6 of=/dev/rst0 bs=10b \ count=10000</pre>
braces { }	Braces indicate required items: <pre>.DEFINE {macro1}</pre>
brackets []	Brackets indicate optional items: <pre>cvtcrt termname [outfile]</pre>
ellipses ...	Ellipses indicate an arbitrary number of similar items: <pre>CHKVAL fieldname value1 value2 ... valueN</pre>
<i>italics</i>	Italic type indicates a variable. Substitute a value for the variable: <pre>library_name</pre>
vertical line	A vertical line indicates a choice within braces or brackets: <pre>FILE filesize [K M]</pre>

Accessing Documentation

The documentation for this release includes platform-specific documentation and generic product documentation.

Platform-Specific Documentation

Platform-specific documentation includes information about installing and using Oracle products on particular platforms. The platform-specific documentation for this product is available in both Adobe portable document format (PDF) and HTML format on the product disc. To access the platform-specific documentation on disc:

1. Use a Web browser to open the `welcome.htm` file in the top-level directory of the disc.
2. For DVD only, select the appropriate product link.
3. Select the **Documentation** tab.

If you prefer paper documentation, then open and print the PDF files.

Product Documentation

Product documentation includes information about configuring, using, or administering Oracle products on any platform. The product documentation for Oracle Database 10g products is available in both HTML and PDF formats in the following locations:

- In the `doc` subdirectory on the Oracle Database 10g DVD

To access the documentation from the DVD, use a Web browser to view the `welcome.htm` file in the top-level directory on the disc, then select the Oracle Database 10g Documentation Library link.

- Online on the Oracle Technology Network (OTN) Web site:
<http://www.oracle.com/technology/documentation/index.html>

Related Documentation

The platform-specific documentation for Oracle Database 10g products includes the following manuals:

- Oracle Database
 - *Oracle Database Release Notes for Apple Mac OS X (Intel)*
 - *Oracle Database Installation Guide for Apple Mac OS X (Intel)*
 - *Oracle Database Administrator's Reference for UNIX-Based Operating Systems*
- Oracle Database Client
 - *Oracle Database Client Installation Guide for Apple Mac OS X (Intel)*
- Oracle Database Companion CD
 - *Oracle Database Companion CD Installation Guide for Apple Mac OS X (Intel)*

Refer to *Oracle Database Release Notes for Apple Mac OS X (Intel)* for important information that was not available when this book was released. The release notes for Oracle Database 10g are updated regularly. You can get the most recent version from Oracle Technology Network at

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

Typographic Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Installation Overview

This chapter describes the different types of Oracle Database installations that you can perform, and issues that you must consider before installing the software. It includes information about the following topics:

- [Overview of Oracle Database Installation](#)
- [Oracle Database Installation Methods](#)
- [Oracle Database Installation Types](#)
- [Database Configuration Options](#)
- [Available Installation Options](#)
- [Installation Considerations](#)

Overview of Oracle Database Installation

The Oracle Database installation process consists of the following phases:

1. **Read the release notes:** Read the Oracle Database 10g release 2 (10.2) release notes before you begin the installation. The release notes are available with the platform-specific documentation. The latest version of the release notes is available on Oracle Technology Network at:
<http://www.oracle.com/technology/documentation>
2. **Planning the installation:** This chapter describes the Oracle products that you can install and issues that you must consider before starting the installation.

You can also refer to [Appendix G](#) which covers frequently asked questions about installing Oracle Database components, such as how to install Oracle Database if the site uses Oracle applications or if you need multiple Oracle Database client connections.
3. **Completing preinstallation tasks:** [Chapter 2](#) describes preinstallation tasks that you must complete before installing the product.
4. **Installing software:** Use the following section to install Oracle Database:
 - [Chapter 3](#) describes how to use Oracle Universal Installer to install Oracle Database.
 - [Appendix A](#) provides information on performing noninteractive (silent) installations, which you may want to use if you must perform multiple installations of Oracle Database.
 - [Appendix B](#) provides information on cloning Oracle home.

- [Appendix E](#) describes globalization support information.
 - [Appendix F](#) provides troubleshooting advice in case you encounter problems with the installation.
 - [Chapter 6](#) describes how to remove Oracle Database.
5. **Completing postinstallation tasks:** [Chapter 4](#) describes recommended and required postinstallation tasks.
 6. **Get started using Oracle Database:** [Chapter 5](#) describes how to check the contents of the installed Oracle Database, how to start various tools, and how to locate various files. [Appendix C](#) describes the Optimal Flexible Architecture, which is a set of guidelines that ensure reliable Oracle installations that require little maintenance, and [Appendix D](#) explains how to manage Oracle Database port numbers.

Oracle Database Installation Methods

You can choose different installation methods to install Oracle Database, as follows:

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

Interactive Installation Methods

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

With Oracle Database 10g release 10.2 on Apple Mac OS X (Intel), Oracle Universal Installer provides two interactive methods that you can use to install Oracle Database:

- **Basic Installation method:**

Select this installation method to quickly install Oracle Database. This installation method requires minimal user input. It installs the software and optionally creates a general-purpose database using the information that you specify on the screen. It is the default installation method.
- **Advanced Installation method:**

Select this installation method to complete any of the following tasks:

 - Perform a custom software installation, or choose a different database configuration

The Available Product Components installation screen automatically selects the components most customers need in their Oracle Database installation. It also lists several components that are not selected by default, but which you may want to include. To find the listing of available components, select **Advanced**, and then in the Installation Type screen, select **Custom**.

 - Select a database character set or different product languages
 - Create the EXAMPLE tablespace during the installation
 - Create a database on a different file system from the software
 - Specify different passwords for administrative schemas
 - Configure automated backups

Automated Installation Methods 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 installation. These automated installation methods are useful if you must perform multiple installations on similarly configured systems or if the system where you want to install the software does not support a graphic user interface.

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

- **Silent Mode:** Oracle Universal Installer runs in silent mode if you use a response file that specifies all required information. None of the Oracle Universal Installer screens are displayed.
- **Suppressed Mode:** Oracle Universal Installer runs in suppressed mode if you do not specify all required information in the response file. Oracle Universal Installer displays only the screens that prompt for the information that you did not specify.

For more information about these modes and about how to complete an installation using response files, refer to [Appendix A](#).

Oracle Database Installation Types

Note: Oracle Database Client is installed separately. You cannot install Oracle Database Client during an Oracle Database installation.

You can choose an installation type from the following list, when installing Oracle Database 10g:

Note: For more information about the features available with each Oracle Database edition and for information about licensing, refer to the *Oracle Database Licensing Information* guide.

- **Enterprise Edition:** Installs licensable Oracle Database options and database configuration and management tools in addition to all of the products that are installed during a Standard Edition installation. It also installs products most commonly used for data warehousing and transaction processing.
- **Standard Edition:** Installs an integrated set of management tools, full distribution, replication, Web features, and facilities for building business-critical applications.

Note: If you perform a Custom installation, then ensure that you install only the components covered by your license. You cannot install Standard Edition using Custom installation.

- **Custom:** Enables you to select the individual components that you want to install from the list of all available components.

Database Configuration Options

During the installation, you can choose whether you want to create an Oracle database as part of the installation. If you choose to create an Oracle database, then Oracle Universal Installer uses Database Configuration Assistant to create it. You can choose to create a preconfigured database type, which is designed for a variety of different applications, or modify a preconfigured database type, or create a customized database to suit your requirements.

This section describes the following database configuration options:

- [Preconfigured Database Types](#)
- [Installation Choices that Affect Database Creation](#)
- [Creating a Database After Installation](#)

Preconfigured Database Types

Oracle provides the following preconfigured database types that you can create or customize during the installation:

- General Purpose
- Transaction Processing
- Data Warehouse

Refer to the online help provided by either Oracle Universal Installer or Database Configuration Assistant for a description of these preconfigured database types.

Installation Choices that Affect Database Creation

Oracle Universal Installer runs Database Configuration Assistant in one of two modes, depending on the choices that you make during the installation:

- Noninteractive mode

If you choose either the Enterprise Edition or Standard Edition installation type, then choose to create a preconfigured database type. Oracle Universal Installer prompts you for the minimum amount of information required to create a database of the type you choose. It then runs Database Configuration Assistant in noninteractive mode to create the database after it installs the software.

- Interactive mode

If you choose the Custom installation type or choose the Advanced database configuration option, then Oracle Universal Installer does not prompt you for database information. Instead, it installs the software and then runs Database Configuration Assistant in interactive mode. Using the screens in Database Configuration Assistant, you can either modify a preconfigured database type or create a custom database and specify precisely how you want to configure it.

Note: If you choose this method to create a database, then click **Help** on any of the Database Configuration Assistant screens for a description of the information that you must specify on that screen.

Creating a Database After Installation

If you decide not to create a database during the installation, then use Database Configuration Assistant to create one after you have installed the software. For more

information about using Database Configuration Assistant to create a database after installation, refer to the *Oracle Database 2 Day DBA* manual.

Available Installation Options

This release of Oracle Database provides [Database Storage Options](#) that you can choose during installation to simplify database administration tasks:

The following sections describe these options.

Database Storage Options

If you choose to create a database during the installation, then specify [File System](#) as the storage options for database files.

File System

If you choose the file system option, then Database Configuration Assistant creates the database files in a directory on a file system mounted on the computer. Oracle recommends that the file system you choose be separate from the file systems used by the operating system or the Oracle software. The file system that you choose can be any of the following:

- A file system on a disk that is physically attached to the system

If you are creating a database on basic disks that are not logical volumes or RAID devices, then Oracle recommends that you follow the Optimal Flexible Architecture (OFA) recommendations described in [Appendix C](#) and distribute the database files over multiple disks.

- A file system on a logical volume manager (LVM) volume or a RAID device

If you are using multiple disks in an LVM or RAID configuration, then Oracle recommends that you use the stripe and mirror everything (SAME) methodology to increase performance and reliability. Using this methodology, you do not need to specify multiple file system mount point for database storage.

If you choose the Custom installation type or the Advanced database creation option, then you can also choose to use the Oracle-managed files feature with the new database. If you use this feature, then you must specify only the database object name instead of file names when creating or deleting database files.

See Also: *Oracle Database Administrator's Guide* for more information about Oracle-managed files

Installation Considerations

This section contains information that you must consider before deciding how to install this product. It contains the following sections:

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

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

<http://metalink.oracle.com>

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

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

Multiple Oracle Homes

This product supports multiple Oracle homes. You can install this release or previous releases of the software multiple times on the same system, in different Oracle home directories.

Installing the Software on a System with an Existing Oracle Installation

You must install this product into a new Oracle home directory. You cannot install products from one release of Oracle Database into an Oracle home directory of a different release. For example, you cannot install release 10.2 software into an existing Oracle9i Oracle home directory. If you attempt to install this release into an Oracle home directory that contains software from an earlier Oracle release, then the installation fails.

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

Preinstallation Tasks

This chapter describes the tasks that you must complete before you start Oracle Universal Installer. It includes information about the following tasks:

- [Logging In to the System as root](#)
- [Checking the Hardware Requirements](#)
- [Checking the Software Requirements](#)
- [Checking the Network Setup](#)
- [Creating Required Operating System Groups and Users](#)
- [Configuring Kernel Parameters](#)
- [Identifying Required Software Directories](#)
- [Identifying or Creating an Oracle Base Directory](#)
- [Choosing a Storage Option for Oracle Database and Recovery Files](#)
- [Creating Directories for Oracle Database or Recovery Files](#)
- [Stopping Existing Oracle Processes](#)
- [Configuring the Oracle User's Environment](#)

Logging In to the System as root

Before you install the Oracle software, you must complete several tasks as the `root` user. To log in as the `root` user, complete the following procedure:

1. Log in to the system as a user that is a member of the `admin` group.
2. Start a local terminal session (`/Applications/Utilities/Terminal`).
3. If you are not logged in as the `root` user, then enter the following command to start a shell session with `root` privileges:

```
$ sudo sh
```
4. When prompted, enter your password.

Note: Alternatively, if you are logged in as a user who is a member of the `admin` group, then use the `sudo` command to run commands as the `root` user, as follows:

```
$ sudo /usr/sbin/sysctl -a
```

Checking the Hardware Requirements

The system must meet the following minimum hardware requirements:

- At least 1 GB of physical RAM
- Amount of swap space configured must be at least the same amount as that of the physical RAM installed
- 400 MB of disk space in the `/tmp` directory
- Between 1.5 GB and 3.5 GB of disk space for the Oracle software, depending on the installation type
- 1.2 GB of disk space for a preconfigured database that uses file system storage (optional)

To ensure that the system meets the hardware requirements, perform the following steps:

1. To determine the physical RAM size, use System Profiler (`/Applications/Utilities/System Profiler`) or enter the following command:

```
# /usr/sbin/system_profiler SPHardwareDataType | grep Memory
```

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

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

```
# df -h /
```

Note: Mac OS X dynamically creates swap files as required in `/private/var/vm` directory. Ensure that you have at least the same amount of disk space as the physical memory installed on the root (`/`) file system to accommodate the swap files that are created.

3. To determine the amount of free disk space on the system, enter the following command:

```
# df -h
```

The following table shows the approximate disk space requirements for software files for each installation type:

Installation Type	Requirement for Software Files (GB)
Enterprise Edition	1.5
Standard Edition	1.5
Custom (max)	2.0

4. To determine whether the system architecture can run the software, enter the following command:

```
# uname -p
```

Note: This command displays the processor type. The command output must be `i386`. If you do not see the expected output, then you cannot install the software on this system.

Checking the Software Requirements

You must verify that the following software is installed on the system. The procedure following the table describes how to verify these requirements.

Item	Requirement
Operating system	Apple Mac OS X Server 10.5.4
Tools	<ul style="list-style-type: none"> ■ Xcode 3.0 or later ■ GNU C compiler (gcc) version 4.0.1 or later This version of gcc is included in Xcode 3.0 or later. ■ JDK 1.4.2_16
Pro*C/C++, Oracle Call Interface, Oracle C++ Call Interface	The version of the GNU C and C++ compiler listed previously is supported for use with these products.
Oracle JDBC/OCI Drivers	<p>You must use the following JDK version with the Oracle JDBC/OCI drivers:</p> <ul style="list-style-type: none"> ■ JDK 1.4.2 with the JNDI extension

To determine the operating system version, enter the following command:

- `# sw_vers`

The output of this command must be similar to the following:

```

ProductName:   Mac OS X Server
ProductVersion: 10.5.4
BuildVersion:  9E17

```

Note: Only the versions listed in the previous table are supported. Do not install the software on other versions of Apple Mac OS X.

Checking the Network Setup

Typically, the computer on which Oracle Database is to be installed, is connected to the network, has local storage to contain the Oracle Database installation, has a display monitor, and has a CD-ROM or DVD drive.

This section describes how to install Oracle Database on computers that do not meet the typical scenario. It covers the following cases:

- [Configuring Name Resolution](#)
- [Installing on DHCP Computers](#)
- [Installing on Multihomed Computers](#)
- [Installing on Computers with Multiple Aliases](#)
- [Installing on Non-Networked Computers](#)

Configuring Name Resolution

When you run Oracle Universal Installer, an error may occur if name resolution is not set up. To avoid this error, before you begin installation, you must ensure that host names are resolved only through the `/etc/hosts` file.

To ensure that host names are resolved only through the `/etc/hosts` file:

1. Verify that the `/etc/hosts` file is used for name resolution. You can do this by using the following command:

```
# dscacheutil -configuration
```

The output of this command must contain `/BSD/local`.

2. Verify that the host name has been set by using the `hostname` command as follows:

```
# hostname
```

The output of this command must be similar to the following:

```
myhost.mycomputer.com
```

3. Verify that the domain name has not been set dynamically by using the `domainname` command as follows:

```
# domainname
```

This command must not return any results.

4. Verify that the hosts file contains the fully qualified host name by using the following command:

```
# cat /etc/hosts | grep `eval hostname`
```

The output of this command must contain an entry for the fully qualified host name and for `localhost`.

For example:

```
192.168.100.16    myhost.us.mycompany.com    myhost
127.0.0.1        localhost                    localhost.localdomain
```

If the hosts file does not contain the fully qualified host name, then open the file and make the required changes in it.

Installing on DHCP Computers

Dynamic Host Configuration Protocol (DHCP) assigns dynamic IP addresses on a network. Dynamic addressing enables a computer to have a different IP address each time it connects to the network. In some cases, the IP address can change while the computer is still connected. You can have a mixture of static and dynamic IP addressing in a DHCP system.

In a DHCP setup, the software tracks IP addresses, which simplifies network administration. This lets you add a new computer to the network without having to manually assign that computer a unique IP address.

Installing on Multihomed Computers

You can install Oracle Database on a multihomed computer. A multihomed computer is associated with multiple IP addresses. This is typically achieved by having multiple

network cards on the computer. Each IP address is associated with a host name. In addition, you can set up aliases for the host name. By default, Oracle Universal Installer uses the `ORACLE_HOSTNAME` environment variable setting to find the host name. If `ORACLE_HOSTNAME` is not set and you are installing on a computer that has multiple network cards, then Oracle Universal Installer determines the host name by using the first entry in the `/etc/hosts` file.

Clients must be able to access the computer either by using this host name or by using aliases for this host name. To verify if clients are able to access the computer, ping the host name from the client computers using the short name (hostname only) and the full name (hostname and domain name). Both tests must be successful.

Setting the `ORACLE_HOSTNAME` Environment Variable

Use the following procedure to set the `ORACLE_HOSTNAME` environment variable.

For example, if the fully qualified host name is `somehost.us.acme.com`, then enter the following commands:

For Bourne, Bash, or Korn shell:

```
$ ORACLE_HOSTNAME=somehost.us.acme.com
$ export ORACLE_HOSTNAME
```

For C shell:

```
% setenv ORACLE_HOSTNAME somehost.us.acme.com
```

Installing on Computers with Multiple Aliases

A computer with multiple aliases is registered with the naming service under a single IP but with multiple aliases. The naming service resolves any of those aliases to the same computer. Before installing Oracle Database on such a computer, set the `ORACLE_HOSTNAME` environment variable to the computer whose host name is to be used.

Installing on Non-Networked Computers

You can install Oracle Database on a non-networked computer. If the computer, such as a laptop, is configured for DHCP and you plan to connect the computer to the network after the Oracle Database installation, then use the `ping` command on the computer on which the database is to be installed, to check if the computer can connect to itself. Perform this step by first using only the host name and then using the fully qualified name, which must be in the `/etc/hosts` file.

Note: When you run the `ping` command on the computer itself, the `ping` command must return the IP address of the computer.

If the `ping` command fails, then contact your network administrator.

Connecting the Computer to the Network after Installation

If you connect the computer to a network after installation, then the Oracle Database instance on your computer can work with other instances on the network. The computer can use a static IP or DHCP, depending on the network to which you are connected.

Creating Required Operating System Groups and Users

Depending on whether this is the first time Oracle software is being installed on this system and on the products that you are installing, you may need to create several operating system groups and users.

The following operating system groups and user are required if you are installing Oracle Database:

- The OSDBA group (`dba`)

You must create this group the first time you install Oracle Database software on the system. It identifies operating system user accounts that have database administrative privileges (the SYSDBA privilege). The default name for this group is `dba`.

To specify a group name other than the default `dba` group, you must choose the Custom installation type to install the software or start Oracle Universal Installer as a user that is not a member of this group. In this case, Oracle Universal Installer prompts you to specify the name of this group.

- The OSOPER group (`oper`)

This is an optional group. Create this group if you want a separate group of operating system users to have a limited set of database administrative privileges (the SYSOPER privilege). By default, members of the OSDBA group also have the SYSOPER privilege.

To specify a separate OSOPER group, other than the default `dba` group, you must choose the Custom installation type to install the software or start Oracle Universal Installer as a user that is not a member of the `dba` group. In this case, Oracle Universal Installer prompts you to specify the name of this group. Usually, `oper` is chosen as the group name.

- An unprivileged user

Verify that the unprivileged user `nobody` exists on the system. The `nobody` user must own the external jobs (`extjob`) executable after the installation.

The following operating system group and user are required for all installations:

- The Oracle Inventory group (typically, `oinstall`)

You must create this group the first time you install Oracle software on the system. The usual name chosen for this group is `oinstall`. This group owns the Oracle inventory, which is a catalog of all Oracle software installed on the system.

Note: If Oracle software is already installed on the system, then the existing Oracle Inventory group must be the primary group of the operating system user that you use to install new Oracle software.

- The Oracle software owner user (typically, `oracle`)

You must create this user the first time you install Oracle software on the system. This user owns all of the software installed during the installation. This user must have the Oracle Inventory group as its primary group. It must also have the OSDBA and OSOPER groups as secondary groups.

Note: In Oracle documentation, this user is referred to as the `oracle` user.

A single Oracle Inventory group is required for all installations of Oracle software on the system. After the first installation of Oracle software, you must use the same Oracle Inventory group for all subsequent Oracle software installations on that system. However, you can choose to create different Oracle software owner users, OSDBA groups, and OSOPER groups (other than `oracle`, `dba`, and `oper`) for separate installations. By using different groups for different installations, members of these different groups have DBA privileges only on the associated databases rather than on all databases on the system.

See Also: *Oracle Database Administrator's Reference for UNIX-Based Operating Systems* and *Oracle Database Administrator's Guide* for more information about the OSDBA and OSOPER groups and the SYSDBA and SYSOPER privileges

Note: The following sections describe how to create local users and groups. As an alternative to creating local users and groups, you could create the appropriate users and groups in a directory service, for example, Network Information Services (NIS). For information about using directory services, contact your system administrator or refer to your operating system documentation.

The following sections describe how to create the required operating system users and groups:

- [Creating the Oracle Inventory Group](#)
- [Creating the OSDBA Group](#)
- [Creating an OSOPER Group \(Optional\)](#)
- [Creating the Oracle Software Owner User](#)

Creating the Oracle Inventory Group

You must create the Oracle Inventory group if it does not exist. The following subsections describe how to determine the Oracle Inventory group name, if it exists, and how to create it if necessary.

Determining Whether the Oracle Inventory Group Exists

When you install Oracle software on the system for the first time, Oracle Universal Installer creates the `oraInst.loc` file. This file identifies the name of the Oracle Inventory group and the path of the Oracle Inventory directory.

To determine whether the Oracle Inventory group exists, enter the following command:

```
# more /var/opt/oracle/oraInst.loc
```

If the output of this command shows the `oinstall` group name, then the group exists.

If the `oraInst.loc` file exists, then the output from this command is similar to the following:

```
inventory_loc=/u01/app/oracle/oraInventory
inst_group=oinstall
```

The `inst_group` parameter shows the name of the Oracle Inventory group, `oinstall`.

Creating the Oracle Inventory Group

If the `oraInst.loc` file does not exist, then create the Oracle Inventory group by using either of the following methods:

- Use Workgroup Manager (`/Applications/Server/Workgroup Manager`) to create a group called `oinstall`.
- Use command-line utilities to create the Oracle Inventory group:
 1. Enter the following command to list existing groups and group IDs (GIDs):

```
# dscl . -list /groups gid name | more
```

2. Identify an unused GID for the new Oracle Inventory group.

3. Enter commands similar to the following to create the `oinstall` group:

```
# dscl . -create /groups/oinstall
# dscl . -append /groups/oinstall gid gid_number
# dscl . -append /groups/oinstall passwd "*"
```

Creating the OSDBA Group

You must create an OSDBA group in the following circumstances:

- An OSDBA group does not exist, for example, if this is the first installation of Oracle Database software on the system
- An OSDBA group exists, but you want to give a different group of operating system users database administrative privileges in a new Oracle installation

If the OSDBA group does not exist or if you require a new OSDBA group, then create it as follows. In the following command, use the group name `dba` unless a group with that name exists.

- Use Workgroup Manager to create a group called `dba`.
- Use command-line utilities to create the OSDBA group:
 1. Enter the following command to list existing groups and GIDs:

```
# dscl . -list /groups gid name | more
```

2. Identify an unused GID for the new OSDBA group.

3. Enter commands similar to the following to create the `dba` group:

```
# dscl . -create /groups/dba
# dscl . -append /groups/dba gid gid_number
# dscl . -append /groups/dba passwd "*"
```

Creating an OSOPER Group (Optional)

Create an OSOPER group only to identify a group of operating system users with a limited set of database administrative privileges (SYSOPER operator privileges). For most installations, it is sufficient to create only the OSDBA group. If you want to use an OSOPER group, then you must create it in the following circumstances:

- If an OSOPER group does not exist, for example, if this is the first installation of Oracle Database software on the system
- If an OSOPER group exists, but you want to give a different group of operating system users database operator privileges in a new Oracle installation

If you require a new OSOPER group, then create it as follows. In the following command, use the group name `oper` unless a group with that name exists.

- Use Workgroup Manager to create a group called `oper`.
- Use command-line utilities to create the OSOPER group:
 1. Enter the following command to list existing groups and GIDs:

```
# dscl . -list /groups gid name | more
```

2. Identify an unused GID for the new OSOPER group.

3. Enter commands similar to the following to create the `oper` group:

```
# dscl . -create /groups/oper
# dscl . -append /groups/oper gid gid_number
# dscl . -append /groups/oper passwd "*"

```

Creating the Oracle Software Owner User

You must create an Oracle software owner user in the following circumstances:

- If an Oracle software owner user does not exist, for example, if this is the first installation of Oracle software on the system
- If an Oracle software owner user exists, but you want to use a different operating system user

Determining Whether an Oracle Software Owner User Exists

To determine whether an Oracle software owner user named `oracle` exists, enter the following command:

```
# id oracle
```

If the `oracle` user exists, then the output from this command is similar to the following:

```
uid=440(oracle) gid=200(oinstall) groups=201(dba),202(oper)
```

If the user exists, then determine whether to use the existing user or create another `oracle` user. To use the existing user, ensure that the user's primary group is the Oracle Inventory group. Refer to one of the following sections for more information:

Note: If necessary, contact your system administrator before using or modifying an existing user.

- To use the existing Oracle software owner user, when the user's primary group is the Oracle Inventory group, refer to the ["Identifying Required Software Directories"](#) on page 2-13.
- To modify an existing user, refer to the ["Modifying an Oracle Software Owner User"](#) on page 2-10.
- To create a user, refer to the following section.

Creating an Oracle Software Owner User

If the Oracle software owner user does not exist or if you require a new Oracle software owner user, then create it as follows. In the following procedure, use the user name `oracle` unless a user with that name exists.

- Use Workgroup Manager to create a user with the following properties:

Property	Recommended Value
Name	Oracle Software Owner
Short name	<code>oracle</code>
Password	Specify and verify a password for the <code>oracle</code> user
Primary group	<code>oinstall</code>
Default shell	Any supported shell, for example <code>/bin/tcsh</code> or <code>/bin/bash</code>
Home directory	A home directory consistent with other user home directories, for example <code>/Users/oracle</code>

- Use command-line utilities to create the Oracle software owner user:
 1. Enter the following command to list existing users and user IDs (UIDs):


```
# dscl . -list /users uid name | more
```
 2. Identify an unused UID for the new Oracle software owner user.
 3. Enter commands similar to the following to create the `oracle` user:


```
# dscl . -create /users/oracle
# dscl . -append /users/oracle uid uid_number
# dscl . -append /users/oracle gid oinstall_gid
# dscl . -append /users/oracle shell /bin/bash
# dscl . -append /users/oracle home /Users/oracle
# dscl . -append /users/oracle realname "Oracle software owner"
```
 4. Create the home directory for the `oracle` user and change the owner and group on that directory:


```
# mkdir /Users/oracle
# chown oracle:oinstall /Users/oracle
```
 5. Set the password for the `oracle` user:


```
# passwd oracle
```

Refer to the ["Identifying Required Software Directories"](#) on page 2-13 to continue.

Modifying an Oracle Software Owner User

If the `oracle` user exists, but its primary group is not `oinstall`, then use either of the following methods to modify it:

- Use Workgroup Manager to modify the `oracle` user, specifying `oinstall` as the Primary group.
- Use command-line utilities to modify the `oracle` user:
 1. Enter the following command to determine the GID of the `oinstall` group:


```
# dscl . -list /groups/oinstall gid name | more
```
 2. Enter the following command to change the GID of the `oracle` user to the GID of the `oinstall` group:


```
# dscl . -delete /users/oracle gid
# dscl . -append /users/oracle gid oinstall_gid
```

Configuring Kernel Parameters

Note: The kernel parameter and shell limit values shown in the following section are recommended values only. For production database systems, Oracle recommends that you tune these values to optimize the performance of the system. Refer to your operating system documentation for more information about tuning kernel parameters.

Verify that the kernel parameters shown in the following table are set to values greater than or equal to the recommended value shown. The procedure following the table describes how to verify and set the values.

Parameter	Value
<code>kern.sysv.semmsl</code>	87381
<code>kern.sysv.semmns</code>	87381
<code>kern.sysv.semmni</code>	87381
<code>kern.sysv.semmnu</code>	87381
<code>kern.sysv.semume</code>	10
<code>kern.sysv.shmall</code>	2097152
<code>kern.sysv.shmmax</code>	Half the size of physical memory (in bytes)
<code>kern.sysv.shmmni</code>	4096
<code>kern.maxfiles</code>	65536
<code>kern.maxfilesperproc</code>	65536
<code>net.inet.ip.portrange.first</code>	1024
<code>net.inet.ip.portrange.last</code>	65000
<code>kern.corefile</code>	core
<code>kern.maxproc</code>	2068
<code>kern.maxproceruid</code>	2068

Note: If the current value for any parameter is higher than the value listed in this table, then do not change the value of that parameter.

To view the current value specified for these kernel parameters, and to change them if necessary:

1. Enter the commands shown in the following table to view the current values of the kernel parameters:

Note: Make a note of the current values and identify any values that you must change.

Parameter	Command
kern.sysv.semmsl, kern.sysv.semms, kern.sysv.semni, kern.sysv.semnu, kern.sysv.semume	# /usr/sbin/sysctl -a grep sem This command displays the value of the semaphore parameters.
kern.sysv.shmall, kern.sysv.shmax, kern.sysv.shmni	# /usr/sbin/sysctl -a grep shm This command displays the details of the shared memory segment sizes.
kern.maxfiles, kern.maxfilesperproc	# /usr/sbin/sysctl -a grep maxfiles This command displays the maximum number of file handles.
net.inet.ip.portrange.first, net.inet.ip.portrange.last	# /usr/sbin/sysctl -a grep net.inet.ip.portrange This command displays the current port number.
kern.corefile	# /usr/sbin/sysctl -a grep corefile
kern.maxproc, kern.maxprocperuid	# /usr/sbin/sysctl -a grep maxproc

2. If the value of any kernel parameter is different from the recommended value, then complete either of the following procedure:

- Use the following command as a root user, to set the values of the kernel parameters:

```
# /usr/sbin/sysctl -w parameter_name=parameter_value
```

- Use any text editor, to create or edit the `/etc/sysctl.conf` file, and add or edit lines similar to the following:

Note: Include lines only for the kernel parameter values to be changed. However, if any of the current values are larger than the recommended value, then specify the larger value.

```
kern.sysv.semmsl=87381
kern.sysv.semms=87381
kern.sysv.semni=87381
```

```
kern.sysv.semnu=87381
kern.sysv.semume=10
kernel.shmall=2097152
kernel.sys.shmmax=2147483648
kernel.sys.shmmni=4096
kern.maxfiles=65536
kern.maxfilesperproc=65536
net.inet.ip.portrange.first=1024
net.inet.ip.portrange.last=65000
kern.corefile=core
kern.maxproc=2068
kern.maxprocperuid=2068
```

If you specify the values in the `/etc/sysctl.conf` file, then the changed values persist when you restart the system.

Setting Shell Limits for the oracle User

To improve the performance of the software on Apple Mac OS X (Intel) systems, you must increase the following shell limits for the `oracle` user:

Shell Limit	Hard Limit
Maximum number of open file descriptors	65536
Maximum number of processes available to a single user	2068

To increase the shell limits:

1. Change directory to the following directory:
2. Open the `IPServices` script using any text editor.
3. Add the following `ulimit` commands to the start of the `StartService` function in this script:

```
ulimit -Hu 2068
ulimit -Su 2068
ulimit -Hn 65536
ulimit -Sn 65536
```

4. Add the same `ulimit` commands to the `/etc/rc` script, before the `SystemStarter` command.

Identifying Required Software Directories

You must identify or create the following directories for the Oracle software:

- [Oracle Base Directory](#)
- [Oracle Inventory Directory](#)
- [Oracle Home Directory](#)

Oracle Base Directory

The Oracle base directory is a top-level directory for Oracle software installations. It is analogous to the `C:\Oracle` directory used for Oracle software installations on Microsoft Windows systems. On Apple Mac OS X (Intel) systems, the Optimal Flexible

Architecture (OFA) guidelines recommend that you use a path similar to the following for the Oracle base directory:

```
/mount_point/app/oracle_sw_owner
```

In this example:

- *mount_point* is the mount point directory for the file system that will contain the Oracle software.
The examples in this guide use `/u01` for the mount point directory. However, you could choose another mount point directory, such as `/oracle` or `/opt/oracle`.
- *oracle_sw_owner* is the operating system user name of the Oracle software owner, for example `oracle`.

You can use the same Oracle base directory for multiple installations or you can create separate Oracle base directories for different installations. If different operating system users install Oracle software on the same system, then each user must create a separate Oracle base directory. The following example Oracle base directories could all exist on the same system:

```
/u01/app/oracle  
/u01/app/orauser  
/opt/oracle/app/oracle
```

The following sections describe how to identify existing Oracle base directories that may be suitable for your installation and how to create an Oracle base directory if necessary.

Regardless of whether you create an Oracle base directory or decide to use an existing one, you must set the `ORACLE_BASE` environment variable to specify the full path to this directory.

Oracle Inventory Directory

The Oracle Inventory directory (`oraInventory`) stores an inventory of all software installed on the system. It is required by, and shared by, all Oracle software installations on a single system. The first time you install Oracle software on a system, Oracle Universal Installer prompts you to specify the path to this directory. Oracle recommends that you choose the following path:

```
oracle_base/oraInventory
```

Oracle Universal Installer creates the directory that you specify and sets the correct owner, group, and permissions for it. You do not need to create it.

Note: All Oracle software installations rely on this directory. Ensure that you back it up regularly.

Do not delete this directory unless you have completely removed all Oracle software from the system.

Oracle Home Directory

The Oracle home directory is the directory where you choose to install the software for a particular Oracle product. You must install different Oracle products, or different releases of the same Oracle product, in separate Oracle home directories. When you run Oracle Universal Installer, it prompts you to specify the path to this directory, and also a name that identifies it. The directory that you specify must be a subdirectory of

the Oracle base directory. Oracle recommends that you specify a path similar to the following for the Oracle home directory:

```
oracle_base/product/10.2.0/db_1
```

Oracle Universal Installer creates the directory path that you specify under the Oracle base directory. It also sets the correct owner, group, and permissions on it. You do not need to create this directory.

Caution: During installation, you must not specify an existing directory that has predefined permissions applied to it as the Oracle home directory. If you do, then you may experience installation failure due to file and group ownership permission errors.

Identifying or Creating an Oracle Base Directory

Before starting the installation, you must either identify an existing Oracle base directory or if required, create one. This section contains information about the following:

- [Identifying an Existing Oracle Base Directory](#)
- [Creating an Oracle Base Directory](#)

Note: You can choose to create an Oracle base directory, even if other Oracle base directories exist on the system.

Identifying an Existing Oracle Base Directory

Existing Oracle base directories may not have paths complying with OFA guidelines. However, if you identify an existing Oracle Inventory directory or existing Oracle home directories, then you can usually identify the Oracle base directories, as follows:

- Identifying an existing Oracle Inventory directory

Enter the following command to view the contents of the `oraInst.loc` file:

```
# more /var/opt/oracle/oraInst.loc
```

If the `oraInst.loc` file exists, then the output from this command is similar to the following:

```
inventory_loc=/u01/app/oracle/oraInventory
inst_group=oinstall
```

The `inventory_loc` parameter identifies the Oracle Inventory directory (`oraInventory`). The parent directory of the `oraInventory` directory is typically an Oracle base directory. In the previous example, `/u01/app/oracle` is an Oracle base directory.

- Identifying existing Oracle home directories

Enter the following command to view the contents of the `oratab` file:

```
# more /etc/oratab
```

If the `oratab` file exists, then it contains lines similar to the following:

```
*:/u03/app/oracle/product/10.2.0/db_1:N
*/opt/orauser/infra_904:N
```

```
*:/oracle/9.2.0:N
```

The directory paths specified on each line identify Oracle home directories. Directory paths that end with the user name of the Oracle software owner to be used are valid choices for an Oracle base directory. If you intend to use the `oracle` user to install the software, then choose either of the following directories from the previous example:

```
/u03/app/oracle
/oracle
```

Note: If possible, choose a directory path similar to the first (`/u03/app/oracle`). This path complies with the OFA guidelines.

Before deciding to use an existing Oracle base directory for this installation, ensure that it satisfies the following conditions:

- It must not be on the same file system as the operating system.
- It must have sufficient free disk space, as follows:

Requirement	Free Disk Space
The Oracle base directory contains only software files.	Up to 3 GB
The Oracle base directory contains both software and database files (not recommended for production databases).	Up to 4 GB

To determine the free disk space on the file system where the Oracle base directory is located, enter the following command:

```
# df -h oracle_base_path
```

To continue:

- If an Oracle base directory exists and you want to use it, then refer to the ["Choosing a Storage Option for Oracle Database and Recovery Files"](#) section on page 2-17.

When you configure the `oracle` user's environment later in this chapter, set the `ORACLE_BASE` environment variable to specify the directory you chose.

- If an Oracle base directory does not exist on the system or if you want to create an Oracle base directory, then refer to the following section.

Creating an Oracle Base Directory

Before you create an Oracle base directory, you must identify an appropriate file system with sufficient free disk space, as follows:

Requirement	Free Disk Space
The Oracle base directory contains only software files.	Up to 3 GB, depending on the platform
The Oracle base directory contains both software and database files (not recommended for production databases).	Up to 4 GB, depending on the platform

To identify an appropriate file system:

1. Use the `df -h` command to determine the free disk space on each mounted file system.
2. From the display, identify a file system that has appropriate free space.
3. Note the name of the mount point directory for the file system that you identified.

To create the Oracle base directory and specify the correct owner, group, and permissions for it:

1. Enter commands similar to the following to create the recommended subdirectories in the mount point directory that you identified and set the appropriate owner, group, and permissions on them:

```
# mkdir -p /mount_point/app/oracle_sw_owner
# chown -R oracle:oinstall /mount_point/app/oracle_sw_owner
# chmod -R 775 /mount_point/app/oracle_sw_owner
```

For example, if the mount point you identify is `/u01` and `oracle` is the user name of the Oracle software owner, then the recommended Oracle base directory path is as follows:

```
/u01/app/oracle
```

2. When you configure the `oracle` user's environment later in this chapter, set the `ORACLE_BASE` environment variable to specify the Oracle base directory that you have created.

Choosing a Storage Option for Oracle Database and Recovery Files

The File System storage option is supported for storing Oracle Database files and Oracle Database recovery files. Oracle Database files include data files, control files, redo log files, the server parameter file, and the password file.

For all installations, you must choose the storage option to be used for Oracle Database files. For enabling automated backups during the installation, you must also choose the storage option for recovery files (the flash recovery area). You do not have to use the same storage option for each file type.

Important: Database files, and recovery files are supported on file systems.

For more information about storage options, refer to the "[Database Storage Options](#)" section on page 1-5.

For information about how to configure disk storage before you start the installation, refer to the following section:

- To use a file system for database or recovery file storage, refer to the "[Creating Directories for Oracle Database or Recovery Files](#)" section on page 2-17.

Creating Directories for Oracle Database or Recovery Files

This section contains the following topics:

- [Guidelines for Placing Oracle Database Files on a File System](#)
- [Guidelines for Placing Oracle Recovery Files on a File System](#)

- [Creating Required Directories](#)

Guidelines for Placing Oracle Database Files on a File System

If you choose to place the Oracle Database files on a file system, then use the following guidelines when deciding where to place them:

- The default path suggested by Oracle Universal Installer for the database file directory is a subdirectory of the Oracle base directory.
- You can choose either a single file system or multiple file systems to store the database files:
 - To use a single file system, choose a file system on a physical device that is dedicated to the database.

For best performance and reliability, choose a RAID device or a logical volume on multiple physical device and implement the stripe-and-mirror-everything (SAME) methodology.
 - To use multiple file systems, choose file systems on separate physical devices that are dedicated to the database.

This method enables you to distribute physical I/O and create separate control files on different devices for increased reliability. It also enables you to fully implement the OFA guidelines described in [Appendix C, "Optimal Flexible Architecture"](#). You must choose either the Advanced database creation option or the Custom installation type during the installation to implement this method.
- To create a preconfigured database during the installation, the file system (or file systems) that you choose must have at least 1.2 GB of free disk space.

For production databases, you must estimate the disk space requirement depending on what the database would be used for.
- For optimum performance, the file systems that you choose must be on physical devices that are used only by the database.
- The `oracle` user must have write permissions to create the files in the path that you specify.

Guidelines for Placing Oracle Recovery Files on a File System

Note: You must choose a location for recovery files only if you intend to enable automated backups during the installation.

If you choose to place the Oracle recovery files on a file system, then use the following guidelines when deciding where to place them:

- To prevent disk failure from making both the database files and the recovery files unavailable, place the recovery files in a file system on a different physical disk from the database files.

The file system that you choose must have at least 2 GB of free disk space.

The disk space requirement is the default disk quota configured for the flash recovery area (specified by the `DB_RECOVERY_FILE_DEST_SIZE` initialization parameter).

If you choose the Custom installation type or the Advanced database configuration option, then you can specify a different disk quota value.

See Also: *Oracle Database Backup and Recovery Basics* for more information about sizing the flash recovery area

- The default path suggested by Oracle Universal Installer for the flash recovery area is a subdirectory of the Oracle base directory.

This default location is not recommended for production databases.

- The oracle user must have write permissions to create the files in the path that you specify.

Creating Required Directories

Note: You must perform this procedure only if you want to place the Oracle Database or recovery files on a separate file system to the Oracle base directory.

To create directories for the Oracle database, or recovery files on separate file systems to the Oracle base directory:

1. Use the `df -h` command to determine the free disk space on each mounted file system.
2. From the display, identify the file systems that to be used:

File Type	File System Requirements
Database files	Choose either: <ul style="list-style-type: none"> ■ A single file system with at least 1.2 GB of free disk space ■ Two or more file systems with at least 1.2 GB of free disk space in total
Recovery files	Choose a file system with at least 2.4 GB of free disk space.

If you are using the same file system for multiple type of files, then add the disk space requirements for each type to determine the total disk space requirement.

3. Note the names of the mount point directories for the file systems that you identified.
4. Enter commands similar to the following to create the recommended subdirectories in each of the mount point directories and set the appropriate owner, group, and permissions on them:

- Database file directory:

```
# mkdir /mount_point/oradata
# chown oracle:oinstall /mount_point/oradata
# chmod 775 /mount_point/oradata
```

- Recovery file directory (flash recovery area):

```
# mkdir /mount_point/flash_recovery_area
# chown oracle:oinstall /mount_point/flash_recovery_area
# chmod 775 /mount_point/flash_recovery_area
```

Stopping Existing Oracle Processes

Caution: If you are installing additional Oracle Database 10g products in an existing Oracle home, then stop *all* processes running in the Oracle home. You must complete this task to enable Oracle Universal Installer to relink certain executables and libraries.

If you choose to create a database during the installation, then most installation types configure and start a default Oracle Net listener using TCP/IP port 1521 and the IPC key value EXTPROC. However, if an existing Oracle Net listener process is using the same port or key value, then Oracle Universal Installer can only configure the new listener; it cannot start it. To ensure that the new listener process starts during the installation, you must shut down any existing listener before starting Oracle Universal Installer.

To determine whether an existing listener process is running and to shut it down if necessary:

1. Switch user to `oracle`:

```
# su - oracle
```

2. Enter the following command to determine whether a listener process is running and to identify its name and the Oracle home directory in which it is installed:

```
$ ps -ef | grep tnslnsr
```

This command displays information about the Oracle Net listeners running on the system:

```
... oracle_home1/bin/tnslnsr LISTENER -inherit
```

In this example, `oracle_home1` is the Oracle home directory where the listener is installed and `LISTENER` is the listener name.

Note: If no Oracle Net listeners are running, then refer to the ["Configuring the Oracle User's Environment"](#) section on page 2-21 to continue.

3. Set the `ORACLE_HOME` environment variable to specify the appropriate Oracle home directory for the listener:

- For Bourne, Bash, or Korn shell:

```
$ ORACLE_HOME=oracle_home1  
$ export ORACLE_HOME
```

- For C or tcsh shell:

```
% setenv ORACLE_HOME oracle_home1
```

4. Enter the following command to identify the TCP/IP port number and IPC key value that the listener is using:

```
$ $ORACLE_HOME/bin/lsnrctl status listenername
```

Note: If the listener uses the default name LISTENER, then you do not have to specify the listener name in this command.

5. Enter a command similar to the following to stop the listener process:

```
$ $ORACLE_HOME/bin/lsnrctl stop listenername
```

6. Repeat this procedure to stop all listeners running on this system.

Configuring the Oracle User's Environment

You run Oracle Universal Installer from the `oracle` account. However, before you start Oracle Universal Installer you must configure the environment of the `oracle` user. To configure the environment, you must:

- Set the default file mode creation mask (`umask`) to 022 in the shell startup file.
- Set the `DISPLAY` environment variable.

To set the `oracle` user's environment:

1. Start a new terminal session.
2. If you are not already logged in to the system where you want to install the software, then log in to that system as the `oracle` user.
3. If you are not logged in as the `oracle` user, then switch user to `oracle`:


```
$ su - oracle
```
4. To determine the default shell for the `oracle` user, enter the following command:


```
$ echo $SHELL
```
5. Open the `oracle` user's shell startup file in any text editor:

Note: On MAC OS X `.bash_profile` is the user startup file for the Bash shell.

- For Bourne shell (`sh`), Bash shell (`bash`), or Korn shell (`ksh`):


```
$ vi .bash_profile
```
 - For C shell (`csh` or `tcsh`):


```
% vi .login
```
6. Enter or edit the following line, specifying a value of 022 for the default file mode creation mask:


```
umask 022
```
 7. If the `ORACLE_SID`, `ORACLE_HOME`, or `ORACLE_BASE` environment variable is set in the file, then remove the appropriate lines from the file.
 8. Save the file, and exit from the editor.
 9. To run the shell startup script, enter the following commands:
 - For Bash shell:

```
$ . ~/.bash_profile
```

- For Bourne or Korn shell:

```
$ . ~/.profile
```

- For C shell:

```
% source ~/.login
```

10. If you determined that the `/tmp` directory has less than 400 MB of free disk space, then identify a file system with at least 400 MB of free space and set the `TMP` and `TMPDIR` environment variables to specify a temporary directory on this file system:

- a. Use the `df -h` command to identify a suitable file system with sufficient free space.

- b. If necessary, enter commands similar to the following to create a temporary directory on the file system that you identify, and then set the appropriate permissions on the directory:

```
$ su - root
# mkdir /mount_point/tmp
# chmod a+wr /mount_point/tmp
# exit
```

- c. Enter commands similar to the following to set the `TMP` and `TMPDIR` environment variables:

- * For Bourne, Bash, or Korn shell:

```
$ TMP=/mount_point/tmp
$ TMPDIR=/mount_point/tmp
$ export TMP TMPDIR
```

- * For C shell:

```
% setenv TMP /mount_point/tmp
% setenv TMPDIR /mount_point/tmp
```

11. Enter commands similar to the following to set the `ORACLE_BASE` and `ORACLE_SID` environment variables:

- For Bourne, Bash, or Korn shell:

```
$ ORACLE_BASE=/u01/app/oracle
$ ORACLE_SID=sales
$ export ORACLE_BASE ORACLE_SID
```

- For C shell:

```
% setenv ORACLE_BASE /u01/app/oracle
% setenv ORACLE_SID sales
```

In these examples, `/u01/app/oracle` is the Oracle base directory that you created or identified earlier and `sales` is the name of the database (typically no more than five characters).

12. Enter the following commands to ensure that the `ORACLE_HOME` and `TNS_ADMIN` environment variables are not set:

- For Bourne, Bash, or Korn shell:

```
$ unset ORACLE_HOME  
$ unset TNS_ADMIN
```

- For C shell:

```
% unsetenv ORACLE_HOME  
% unsetenv TNS_ADMIN
```

Note: If the ORACLE_HOME environment variable is set, then Oracle Universal Installer uses the value that it specifies as the default path for the Oracle home directory. However, if you set the ORACLE_BASE environment variable, then Oracle recommends that you unset the ORACLE_HOME environment variable and choose the default path suggested by Oracle Universal Installer.

13. To verify that the environment has been set correctly, enter the following commands:

```
$ umask  
$ env | more
```

Verify that the umask command displays a value of 22, 022, or 0022 and the environment variables that you set in this section have the correct values.

Installation Tasks

The Oracle Database software is available on disc (DVD) or you can download it from the Oracle Technology Network Web site. In most cases, you use the graphical user interface (GUI) provided by Oracle Universal Installer to install the software. However, you can also use Oracle Universal Installer to complete silent-mode installations, without using the GUI.

See Also: [Appendix A](#) for information about silent-mode installations

This chapter includes information about the following topics:

- [Preinstallation Considerations](#)
- [Reviewing Component-Specific Installation Guidelines](#)
- [Accessing the Installation Software](#)
- [Installing the Oracle Database Software](#)

Preinstallation Considerations

After reviewing the information in [Chapter 1, "Installation Overview"](#) and completing the tasks listed in [Chapter 2, "Preinstallation Tasks"](#), consider the following issues:

Performing Multiple Oracle Database Installations in Noninteractive Mode

If you must perform multiple installations of Oracle Database, then you can use noninteractive mode. In noninteractive mode, at each node, you run Oracle Universal Installer from the command line using a response file. The response file is a text file containing the settings you normally enter in the Oracle Universal Installer GUI dialog boxes.

See Also: [Appendix A](#) for information about silent-mode installations

Reviewing Component-Specific Installation Guidelines

Review the following guidelines before starting Oracle Universal Installer:

- Oracle Universal Installer

Do not use Oracle Universal Installer from an earlier Oracle release to install components from this release.

- Products not installed by default: choose **Advanced Installation** and then the **Custom** installation type. These products are:
 - Oracle Connection Manager
 - Oracle Label Security

To configure Oracle Label Security to use Oracle Internet Directory, choose the Oracle Internet Directory option when running Database Configuration Assistant. If you are installing Oracle Label Security in an existing Oracle home, then shut down each database in the Oracle home.
 - Data Mining Scoring Engine
- Reinstalling Oracle Software

If you reinstall Oracle software into an Oracle home directory where Oracle Database is already installed, then you must also reinstall any components, such as Oracle Partitioning, that were installed before you began the reinstallation.

Accessing the Installation Software

The Oracle Database software is available on compact disc (DVD) or you can download it from the Oracle Technology Network Web site. To install the software from the hard disk, you must either download it from Oracle Technology Network and unpack it, or copy it from the disc, if you have it.

You can access and install Oracle Database by using one of the following methods:

Note: If the Oracle Database 10g software is on two discs, then you can avoid having to change discs during the installation by running Oracle Universal Installer from the DVD or by copying the contents of the installation discs to a hard disk.

- To copy the software to a hard disk, refer to the following section: "[Copy the Oracle Database Software to a Hard Disk](#)" on page 3-3
- To install the software from a disc or from an existing hard disk location, refer to the "[Installing the Oracle Database Software](#)" section on page 3-4.
- To download the software from Oracle Technology Network, refer to the following section.

Downloading and Extracting Oracle Software from the OTN Web Site

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

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

Downloading the Installation Archive Files

To download the installation archive files from Oracle Technology Network:

1. Use any browser to access the software download page from Oracle Technology Network:

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

2. Navigate to the download page for the product to be installed.
3. On the 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 archive files.
In most cases, the available disk space must be at least twice the size of all of the archive files.
5. On the file system that you selected in step 4, create a parent directory for each product, for example OraDB10g, to hold the installation directories.
6. Download all of the installation archive files to the directory that you created in step 5.
7. Verify that the files you downloaded are the same size as the corresponding files on Oracle Technology Network.

Extracting the Installation Files

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

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

```
unzip filename.zip
```

When you have extracted all of the required installation files, refer to the following section:

["Installing the Oracle Database Software"](#) on page 3-4

Copy the Oracle Database Software to a Hard Disk

If the software for Oracle Database 10g requires two discs on your platform, then you can avoid having to mount and unmount discs during the installation by copying the contents of each CD-ROM to a hard disk on the system, or by installing the software from the DVD-ROM.

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

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

```
$ mkdir OraDb10g
```

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

```
$ cd OraDb10g
```

3. Create subdirectories for each installation disc with names similar to the following, where *n* is the disc number:

```
$ mkdir Diskn
```

4. Insert the first disc into the disc drive.
5. Copy the contents of the mounted disc to the corresponding new subdirectory as follows:

```
$ cp -R /directory_path Diskn
```

In this example, *n* is the number of the installation CD-ROM and */directory_path* is the CD-ROM mount point directory or the path of the db directory on the DVD-ROM.

6. If necessary, mount the next disc and repeat step 5.

After you have copied all of the required discs, see the "[Installing the Oracle Database Software](#)" section.

Installing the Oracle Database Software

The following sections describe how to install the Oracle software:

- [Reviewing Product-Specific Installation Guidelines](#)
- [Running the Oracle Universal Installer](#)

Reviewing Product-Specific Installation Guidelines

Review the following guidelines before starting the Installer:

- Oracle Universal Installer

Do not use Oracle Universal Installer from an earlier Oracle product release to install components from this release.

- Products requiring a custom installation

To install the following products, you must choose the Custom installation type:

Apply the following guidelines when specifying passwords:

- Oracle Label Security

Note: Before you perform a custom installation, ensure that all databases in the Oracle home where Oracle Label Security is to be installed are shut down.

If you want to configure OLS to use Oracle Internet Directory, then you must choose the Oracle Internet Directory option when running the DBCA.

- Oracle Connection Manager
- Data Mining Scoring Engine

- Re-installing Oracle software

If you re-install Oracle software into an Oracle home directory where Oracle Database is already installed, then you must also re-install any components, such as Oracle Partitioning, that were installed before you began the re-installation.

Running the Oracle Universal Installer

This section describes the Basic Installation as a default setting. For any type of installation process, start Oracle Universal Installer and install the software, as follows:

1. If you are installing the software from disc, then mount the first disc if it is not already mounted.
2. To start Oracle Universal Installer, complete one of the following steps depending on the location of the installation files:

Note: Start Oracle Universal Installer from the terminal session where you logged in as the `oracle` user and set the user's environment (described in [Chapter 2](#)).

- If the installation files are on disc, then enter commands similar to the following, where `directory_path` is the path of the database directory on the DVD:

```
$ cd /tmp
$ /directory_path/runInstaller
```

- If the installation files are on the hard disk, then change directory to the `client` directory and enter the following command:

```
$ ./runInstaller
```

3. Use the following guidelines to complete the installation:

- Follow the instructions displayed on the Oracle Universal Installer screens. If you need additional information, then click **Help**.
- When Oracle Universal Installer prompts you to run a script with `root` privileges, enter a command similar to the following in a terminal where you are logged in as the `root` user, then click **Continue** or **OK**:

```
# /script_path/script_name
```

Alternatively, enter a command similar to the following to run the script in a terminal where you are logged in as a user that is a member of the `admin` group:

```
$ sudo /script_path/script_name
```

The following table lists the various screens displayed and the recommended actions required when you select the Basic Installation option.

Screen	Recommended Action
Select Installation Method	<p>The Basic Installation option is selected by default.</p> <p>Specify the directory path of the Oracle home. Ensure that the UNIX DBA group <code>oinstall</code> is selected. If a starter database is to be created, then specify a name and password for it. Then, click Next.</p> <p>Note: This screen consists of Basic Installation and Advanced Installation options. Select the Advanced Installation option if you intend to use custom installation.</p>
Specify Inventory Directory and Credentials	<p>This screen is displayed only during the first installation of Oracle products on a system.</p> <p>Specify the full path of the Oracle Inventory directory. Ensure that the operating system group selected is <code>oinstall</code>. Then, click Next.</p>

Screen	Recommended Action
Product-Specific Prerequisite Checks	<p>Verify that all of the prerequisite checks succeed, then click Next.</p> <p>Oracle Universal Installer checks the system to verify that it is configured correctly to run Oracle software. If you have completed all of the preinstallation steps in this guide, then all the checks must pass.</p> <p>If a check fails, then review the cause of the failure listed for that check on the screen. If possible, rectify the problem and rerun the check. Alternatively, if you are satisfied that your system meets the requirements, then you can select the check box for the failed check to manually verify the requirement.</p>
Summary	Review the information displayed on this screen, and then click Install .
Install	This screen displays status information while the product is being installed.
Configuration Assistants	This screen displays status information for the configuration assistants that configure the software and create a database. When the message is displayed after completion of the DBCA process, click OK to continue.
Execute Configuration Scripts	When prompted, read the instructions and then run the scripts mentioned on this screen. Click OK to continue.
End of Installation	<p>The configuration assistants configure several Web-based applications. This screen displays the URLs configured for these applications. Make a note of the URLs used. The port numbers used in these URLs are also recorded in the following file:</p> <p><code>oracle_home/install/portlist.ini</code></p> <p>To exit from Oracle Universal Installer, click Exit and then click Yes.</p>

- When prompted for the SYS, SYSTEM, SYSMAN, and DBSNMP passwords, Oracle recommends that you specify passwords that:
 - Are at least four characters long
 - Are different from the user names
 - Have at least one alphabetic, one numeric, and one punctuation mark character
 - Are not simple or obvious words, such as welcome, account, database, or user

Note: You must remember the passwords that you specify.

- If you encounter errors while installing or linking the software, then refer to [Appendix F](#) for information about troubleshooting.
4. If you chose an installation type that runs Database Configuration Assistant and Oracle Net Configuration Assistant (NetCA) in interactive mode, then you must provide detailed information about configuring the database and network.

If you need assistance when using the Database Configuration Assistant or NetCA in interactive mode, then click **Help** on the screen.

Note: If you chose a default installation, then Database Configuration Assistant and NetCA run in noninteractive mode.

5. When all of the configuration tools have finished, click **Exit**, then click **Yes** to exit from Oracle Universal Installer.

Postinstallation Tasks

This chapter describes how to complete postinstallation tasks after you have installed the software. It includes information about the following sections:

- [Required Postinstallation Tasks](#)
- [Recommended Postinstallation Tasks](#)
- [Required Product-Specific Postinstallation Tasks](#)
- [Installing Oracle Database 10g Products from the Companion CD](#)

You must perform the tasks listed in the "[Required Postinstallation Tasks](#)" section. Oracle recommends that you perform the tasks listed in the "[Recommended Postinstallation Tasks](#)" section after all installations.

If you installed and intend to use any of the products listed in the "[Required Product-Specific Postinstallation Tasks](#)" section, then you must perform the tasks listed in the product-specific subsections.

Note: This chapter describes basic configuration only. Refer to *Oracle Database Administrator's Reference for UNIX-Based Operating Systems* and product-specific administration and tuning guides for more detailed configuration and tuning information.

Required Postinstallation Tasks

You must perform the tasks described in the following sections after completing an installation:

- [Downloading and Installing Patches](#)
- [Configuring Oracle Products](#)

Downloading and Installing Patches

Check the *OracleMetaLink* Web site for required patches for the installation.

To download required patches:

1. Use a Web browser to view the *OracleMetaLink* Web site:
<http://metalink.oracle.com>
2. Log in to *OracleMetaLink*.

Note: If you are not an Oracle*MetaLink* registered user, then click **Register for MetaLink!** and follow the registration instructions.

3. On the main Oracle*MetaLink* page, click **Patches**.
4. Select **Simple Search**.
5. Specify the following information, then click **Go**:
 - In the **Search By** field, choose Product or Family, then specify RDBMS Server.
 - In the **Release** field, specify the current release number.
 - In the **Patch Type** field, specify Patchset/Minipack.
 - In the **Platform or Language** field, select your platform.

Configuring Oracle Products

Many Oracle products and options must be configured before you use them for the first time. Before using individual Oracle products or options, refer to the appropriate manual in the product documentation library.

Recommended Postinstallation Tasks

Oracle recommends that you perform the tasks described in the following section after completing an installation:

- [Creating a Backup of the root.sh Script](#)
- [Configuring New or Upgraded Databases](#)
- [Setting Up User Accounts](#)
- [Setting the NLS_LANG Environment Variable](#)
- [Generating the Client Static Library](#)

Creating a Backup of the root.sh Script

Oracle recommends that you back up the `root.sh` script after you complete an installation. If you install other products in the same Oracle home directory, then Oracle Universal Installer updates the contents of the existing `root.sh` script during the installation. If you require information contained in the original `root.sh` script, then you can recover it from the backed up `root.sh` file.

Configuring New or Upgraded Databases

Oracle recommends that you run the `utlrbp.sql` script after creating or upgrading a database. This script recompiles all PL/SQL modules that might be in an invalid state, including packages, procedures, and types. This is an optional step but Oracle recommends that you do it during installation and not at a later date.

To run the `utlrbp.sql` script, follow these steps:

1. Switch user to `oracle`.
2. Use the `oraenv` or `coraenv` script to set the environment for the database where you want to run the `utlrbp.sql` script:
 - For Bourne, Bash, or Korn shell:

```
$ . /usr/local/bin/oraenv
```

- For C shell:

```
% source /usr/local/bin/coraenv
```

When prompted, specify the SID for the database.

3. Start SQL*Plus, as follows:

```
$ sqlplus "/ AS SYSDBA"
```

4. If necessary, start the database:

```
SQL> STARTUP
```

5. Run the `utlrp.sql` script:

```
SQL> @?/rdms/admin/utlrp.sql
```

Setting Up User Accounts

For information about setting up additional user accounts, refer to *Oracle Database Administrator's Reference for UNIX-Based Operating Systems*.

Setting the NLS_LANG Environment Variable

NLS_LANG is an environment variable that specifies the locale behavior for Oracle software. This variable sets the language and territory used by the client application and the database server. It also sets the character set for the client, which is the character set for data entered or displayed by an Oracle client program, such as SQL*Plus.

See Also: [Appendix E, "Configuring Oracle Database Globalization Support"](#) for more information about the NLS_LANG environment variable

Generating the Client Static Library

The client static library (`libclntst10.a`) is not generated during installation. If you want to link your applications to the client static library, then you must first generate it as follows:

1. Switch user to `oracle`.
2. Set the `ORACLE_HOME` environment variable to specify the Oracle home directory used by the Oracle Database installation. For example:

- For Bourne, Bash, or Korn shell:

```
$ ORACLE_HOME=/u01/app/oracle/product/10.2.0/db_1
$ export ORACLE_HOME
```

- For C shell:

```
% setenv ORACLE_HOME /u01/app/oracle/product/10.2.0/db_1
```

3. Enter the following command:

```
$ $ORACLE_HOME/bin/genclntst
```

Required Product-Specific Postinstallation Tasks

The following sections describe platform-specific postinstallation tasks that you must perform if you install and intend to use the products mentioned:

- [Configuring Oracle Net Services](#)
- [Configuring Oracle Label Security](#)
- [Installing Natively Compiled Java Libraries for Oracle JVM and Oracle interMedia](#)
- [Installing Oracle Text Supplied Knowledge Bases](#)
- [Configuring Oracle Messaging Gateway](#)
- [Configuring Oracle Precompilers](#)
- [Configuring OCCI](#)

Note: You need only perform postinstallation tasks for products that you intend to use.

Configuring Oracle Net Services

If you have a previous release of Oracle software installed on this system, then you can copy information from the Oracle Net `tnsnames.ora` and `listener.ora` configuration files from the previous release to the corresponding files for the new release.

Note: The default location for the `tnsnames.ora` and `listener.ora` files is the `$ORACLE_HOME/network/admin/` directory. However, you can also use a central location for these files, for example `/etc`.

Modifying the `listener.ora` File

If you are upgrading from a previous release of Oracle Database, then Oracle recommends that you use the current release of Oracle Net listener instead of the listener from the previous release.

To use the listener from the current release, you may need to copy static service information from the `listener.ora` file from the previous release to the version of that file used by the new release.

For any database instances earlier than release 8.0.3, add static service information to the `listener.ora` file. Oracle Database releases later than release 8.0.3 do not require static service information.

Modifying the `tnsnames.ora` File

Unless you are using a central `tnsnames.ora` file, copy Oracle Net service names and connect descriptors from the previous release `tnsnames.ora` file to the version of that file used by the new release.

If necessary, you can also add connection information for additional database instances to the new file.

Configuring Oracle Label Security

If you installed Oracle Label Security, then you must configure it in a database before you use it. You can configure Oracle Label Security in two ways; with Oracle Internet Directory integration and without Oracle Internet Directory integration. If you configure Oracle Label Security without Oracle Internet Directory integration, then you cannot configure it to use Oracle Internet Directory at a later stage.

Note: To configure Oracle Label Security with Oracle Internet Directory integration, Oracle Internet Directory must be installed in your environment and the Oracle database must be registered in the directory.

See Also: *Oracle Label Security Administrator's Guide* for more information about Oracle Label Security enabled with Oracle Internet Directory

Installing Natively Compiled Java Libraries for Oracle JVM and Oracle *interMedia*

If you plan to use Oracle JVM or Oracle *interMedia*, then Oracle strongly recommends that you install the natively compiled Java libraries (NCOMPs) used by those products from the Oracle Database 10g Companion CD. These libraries are required to improve the performance of the products on your platform.

See Also: For information about how to install products from the Companion CD, refer to the "[Installing Oracle Database 10g Products from the Companion CD](#)" section on page 4-8

Installing Oracle Text Supplied Knowledge Bases

An Oracle Text knowledge base is a hierarchical tree of concepts used for theme indexing, ABOUT queries, and deriving themes for document services. If you plan to use any of these Oracle Text features, then you can install two supplied knowledge bases (English and French) from the Oracle Database 10g Companion CD.

Note: You can extend the supplied knowledge bases depending on your requirements. Alternatively, you can create your own knowledge bases, possibly in languages other than English and French.

See Also:

- *Oracle Text Reference* for more information about creating and extending knowledge bases
- The "[Installing Oracle Database 10g Products from the Companion CD](#)" section on page 4-8 for information about how to install products from the Companion CD

Configuring Oracle Messaging Gateway

To configure Oracle Messaging Gateway, refer to the section about Messaging Gateway in *Oracle Streams Advanced Queuing User's Guide and Reference*. When following the instructions listed in that manual, refer to this section for additional

instructions about configuring the `listener.ora`, `tnsnames.ora`, and `mgw.ora` files.

Modifying the listener.ora File for External Procedures

To modify the `$ORACLE_HOME/network/admin/listener.ora` file for external procedures:

1. Back up the `listener.ora` file.
2. Ensure that the default IPC protocol address for external procedures is set as follows:


```
(ADDRESS = (PROTOCOL=IPC) (KEY=EXTPROC))
```
3. Add static service information for a service called `mgwextproc` by adding lines similar to the following to the `SID_LIST` parameter for the listener in the `listener.ora` file:

```
(SID_DESC =
  (SID_NAME = mgwextproc)
  (ENVS = EXTPROC_DLLS=/oracle_home/lib32/libmgwagent.dylib,
  DYLD_LIBRARY_
  PATH=/System/Library/Frameworks/JavaVM.framework/Versions/1.4.2/Libraries:/orac
  le_home/lib32
  (ORACLE_HOME = oracle_home)
  (PROGRAM = extproc32)
  )
```

In this example:

- The `ENVS` parameter defines the shared library path environment variable and any other required environment variables.

In the settings for the shared library path environment variable, you must also add any additional library paths required for non-Oracle messaging systems, for example WebSphere MQ or TIBCO Rendezvous.

- `oracle_home` is the path of the Oracle home directory.
- `extproc32` is the external procedure agent executable file

The following example shows a sample `listener.ora` file:

Note: In the following example, the `PLSExtProc` service is the default service for PL/SQL external procedures.

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PLSExtProc)
      (ORACLE_HOME = /u01/app/oracle/product/10.2.0/db_1)
      (PROGRAM = extproc)
    )
    (SID_DESC =
      (SID_NAME = mgwextproc)
      (ENVS = EXTPROC_DLLS=/u01/app/oracle/product/10.2.0/db_
      1/lib32/libmgwagent.dylib,
      DYLD_LIBRARY_PATH
      =/System/Library/Frameworks/JavaVM.framework/Versions/1.4.2/Libraries:
      /u01/app/oracle/product/10.2.0/db_1/lib32)
```

```

        (ORACLE_HOME = /u01/app/oracle/product/10.2.0/db_1)
        (PROGRAM = extproc32)
    )
)

```

Modifying the tnsnames.ora File for External Procedures

To modify the `$ORACLE_HOME/network/admin/tnsnames.ora` file for external procedures:

1. Back up the `tnsnames.ora` file.
2. In the `tnsnames.ora` file, add a connect descriptor with the net service name `MGW_AGENT`, as follows:

```

MGW_AGENT =
(DESCRIPTION=
  (ADDRESS_LIST= (ADDRESS= (PROTOCOL=IPC) (KEY=EXTPROC)))
  (CONNECT_DATA= (SID=mgwextproc) (PRESENTATION=RO)))

```

In this example:

- The value specified for the `KEY` parameter must match the value specified for that parameter in the IPC protocol address in the `listener.ora` file.
- The value of the `SID` parameter must match the service name in the `listener.ora` file that you specified for the Oracle Messaging Gateway external procedure agent in the previous section (`mgwextproc`).

Setting Up the mgw.ora Initialization File

To modify the `$ORACLE_HOME/mgw/admin/mgw.ora` file for external procedures, set the `CLASSPATH` environment variable to include the classes in the following table and any additional classes required for Oracle Messaging Gateway to access non-Oracle messaging systems, for example WebSphere MQ or TIBCO Rendezvous classes:

Classes	Path
Oracle Messaging Gateway	<code>ORACLE_HOME/mgw/classes/mgw.jar</code>
JRE internationalization	<code>ORACLE_HOME/jdk/JRE/lib/i18n.jar</code>
JRE runtime	<code>ORACLE_HOME/jdk/JRE/lib/rt.jar</code>
Oracle JDBC	<code>ORACLE_HOME/jdbc/lib/ojdbc14.jar</code>
Oracle internationalization	<code>ORACLE_HOME/jdbc/lib/orai18n.jar</code>
SQLJ	<code>ORACLE_HOME/oc4j/sqlj/lib/translator.jar</code> <code>ORACLE_HOME/sqlj/lib/runtime12.tar</code>
JMS Interface	<code>ORACLE_HOME/rdbms/jlib/jmscommon.jar</code>
Oracle JMS implementation	<code>ORACLE_HOME/rdbms/jlib/aqapil3.jar</code>
Java Transaction API	<code>ORACLE_HOME/jlib/jta.jar</code>

Note: All the lines in the `mgw.ora` file must consist of less than 1024 characters.

Configuring Oracle Precompilers

This section describes postinstallation tasks for Oracle precompilers.

Note: All precompiler configuration files are located in the `ORACLE_HOME/precomp/admin` directory.

Configuring Pro*C/C++

Verify that the `PATH` environment variable setting includes the directory that contains the C compiler executable. The default directory for the `gcc` compiler executable is `/usr/bin`.

To verify the path setting of the `gcc` compiler, you must use the following command:

```
$ which gcc
```

Configuring OCCI

The Oracle C++ Call Interface (OCCI) libraries for use with GNU C++ compiler version 4.0.1 are installed with this release. The libraries are installed in the following locations:

```
ORACLE_HOME/lib/libocci10.a  
ORACLE_HOME/lib/libocci10.dylib.10.1
```

Installing Oracle Database 10g Products from the Companion CD

The Oracle Database 10g Companion CD contains additional products that you can install. Whether you must install these products depends on which Oracle Database products or features you plan to use. If you plan to use the following products or features, then you must complete the Oracle Database 10g Products installation from the Companion CD:

- JPublisher
- Oracle JVM
- Oracle *interMedia*
- Oracle JDBC development drivers
- Oracle SQLJ
- Oracle Database Examples
- Oracle Text supplied knowledge bases
- Oracle Ultra Search

For information about installing software from the Companion CD, refer to *Oracle Database Companion CD Installation Guide for Apple Mac OS X (Intel)*.

Getting Started with Oracle Database

This chapter provides information about the default preconfigured database, including information about Oracle database accounts, passwords, and file locations. It includes information about the following topics:

- [Checking the Installed Oracle Database Contents and Directory Location](#)
- [Reviewing Accounts and Passwords](#)
- [Unlocking and Resetting User Passwords](#)
- [Identifying Databases](#)
- [Locating the Server Parameter File](#)
- [Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files](#)

Checking the Installed Oracle Database Contents and Directory Location

You can use Oracle Universal Installer to check the contents and directory location of an Oracle Database installation. To do this:

1. Start Oracle Universal Installer, follow the instructions in "[Running the Oracle Universal Installer](#)" on page 3-4.
2. Click **Installed Products** to display the Inventory dialog box on the Welcome screen.
3. Select the Oracle Database product from the list to check the installed contents.
4. Click **Details** to find additional information about an installed product.
5. Click the **Environment** tab to check the directory location of the installed contents.
6. Click **Close** to close the Inventory dialog box.
7. Click **Cancel** to close Oracle Universal Installer, and then click **Yes** to confirm.

Reviewing Accounts and Passwords

All databases created by the Database Configuration Assistant (DBCA) include the SYS, SYSTEM, SYSMAN, and DBSNMP database accounts. In addition, Oracle provides several other administrative accounts. Before using these other accounts, you must unlock them and reset their passwords. [Table 5-1](#) describes these accounts, listing their user names and default passwords.

See Also: The "[Unlocking and Resetting User Passwords](#)" section on page 5-3 for information about unlocking and resetting passwords

Table 5–1 Database Accounts

User Name	Default Password	Description	See Also
ANONYMOUS	ANONYMOUS	Allows HTTP access to Oracle XML DB.	None
BI	BI	The account that owns the Business Intelligence schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
CTXSYS	CTXSYS	The Oracle Text account.	<i>Oracle Text Reference</i>
DIP	DIP	The account used by the Directory Integration Platform (DIP) to synchronize the changes in Oracle Internet Directory with the applications in the database.	<i>Oracle Internet Directory Administrator's Guide</i>
DMSYS	DMSYS	The data mining account. DMSYS performs data mining operations.	<i>Oracle Data Mining Administrator's Guide</i>
EXFSYS	EXFSYS	The account that owns the Expression Filter schema.	None
HR	HR	The account that owns the Human Resources schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
IX	IX	The account that owns the Information Transport schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
LBACSYS	LBACSYS	The Oracle Label Security administrator account.	<i>Oracle Label Security Administrator's Guide</i>
MDDATA	MDDATA	The schema used by Oracle Spatial for storing Geocoder and router data.	<i>Oracle Spatial User's Guide and Reference</i>
MDSYS	MDSYS	The Oracle Spatial and Oracle <i>interMedia</i> Locator administrator account.	<i>Oracle Spatial User's Guide and Reference</i>
OE	OE	The account that owns the Order Entry schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
OLAPSYS	MANAGER	The account that owns the OLAP catalogs.	<i>Oracle OLAP Application Developer's Guide</i>
ORDPLUGINS	ORDPLUGINS	The Oracle <i>interMedia</i> user. Plugins supplied by Oracle and third-party plugins are installed in this schema.	<i>Oracle interMedia Reference</i>
ORDSYS	ORDSYS	The Oracle <i>interMedia</i> administrator account.	<i>Oracle interMedia Reference</i>

Table 5–1 (Cont.) Database Accounts

User Name	Default Password	Description	See Also
OUTLN	OUTLN	The account that supports plan stability. Plan stability enables you to maintain the same execution plans for the same SQL statements. OUTLN acts as a role to centrally manage metadata associated with stored outlines.	<i>Oracle Database Concepts</i>
PM	PM	The account that owns the Product Media schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	<i>Oracle Database Sample Schemas</i>
SCOTT	TIGER	An account used by Oracle sample programs and examples.	<i>Oracle Database Administrator's Guide</i>
SH	SH	The account that owns the Sales History schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas during an Enterprise Edition installation.	<i>Oracle Database Administrator's Guide</i>
SI_INFORMTN_SCHEMA	SI_INFORMTN_SCHEMA	The account that stores the information views for the SQL/MM Still Image Standard.	<i>Oracle interMedia Reference</i>
SYS	Password set during installation or in the Database Configuration Assistant	The account used to perform database administration tasks.	<i>Oracle Database Administrator's Guide</i>
SYSTEM	Password set during installation or in the Database Configuration Assistant	Another account used to perform database administration tasks.	<i>Oracle Database Administrator's Guide</i>
WMSYS	WMSYS	The account used to store the metadata information for Oracle Workspace Manager.	<i>Oracle Database Application Developer's Guide - Workspace Manager</i>
WKPROXY	WKSYS	The Ultra Search proxy user.	<i>Oracle Ultra Search Administrator's Guide</i>
WK_TEST	WK_TEST	The default Ultra Search instance schema.	<i>Oracle Ultra Search Administrator's Guide</i>
WKSYS	WKSYS	The account used to store Ultra Search system dictionaries and PL/SQL packages.	<i>Oracle Ultra Search Administrator's Guide</i>
XDB	XDB	The account used for storing Oracle XML DB data and metadata.	<i>Oracle XML DB Developer's Guide</i>

Unlocking and Resetting User Passwords

Passwords for all Oracle system administration accounts except SYS, SYSTEM, SYSMAN, and DBSMP are revoked after installation. Before you use a locked account,

you must unlock it and reset its password. If you created a preconfigured database during the installation, but you did not unlock a required account, then you must unlock it now, using the following method:

- [Using SQL*Plus to Unlock Accounts and Reset Passwords](#)

Note: If you are creating a database using Database Configuration Assistant, then you can unlock accounts after the database is created by clicking **Password Management** before you exit from Database Configuration Assistant.

Using SQL*Plus to Unlock Accounts and Reset Passwords

To unlock and reset user account passwords using SQL*Plus:

1. Start SQL*Plus and log in as the SYS user, connecting as SYSDBA:

```
$ sqlplus /nolog
SQL> CONNECT SYS/SYS_password AS SYSDBA
```

2. Enter a command similar to the following, where *account* is the user account to be unlocked and *password* is the new password:

```
SQL> ALTER USER account [ IDENTIFIED BY password ] ACCOUNT UNLOCK;
```

In this example:

- The `ACCOUNT UNLOCK` clause unlocks the account.
- The `IDENTIFIED BY password` clause resets the password.

Note: If you unlock an account but do not reset the password, then the password remains expired. The first time someone connects as that user, they must change the user's password.

To permit unauthenticated access to your data through HTTP, unlock the ANONYMOUS user account.

See Also: *Oracle Database Administrator's Guide* for more information about:

- Unlocking and changing passwords after installation
- Oracle security procedures
- Best security practices

Identifying Databases

The Oracle Database 10g software identifies a database by its global database name. A global database name consists of the database name and database domain. Usually, the database domain is the same as the network domain, but it need not be. The global database name uniquely distinguishes a database from any other database in the same network. You specify the global database name when you create a database during the installation, or using the Database Configuration Assistant. For example:

```
sales.us.example.com
```

In this example:

- `sales` is the name of the database. The database name portion is a string of no more than 30 characters that can contain alphanumeric, underscore (`_`), dollar (`$`), and pound (`#`) characters. The `DB_NAME` initialization parameter specifies the database name.
- `us.example.com` is the database domain in which the database is located. In this example, the database domain is the same as the network domain. Together, the database name and the database domain make the global database name unique. The domain portion is a string of no more than 128 characters that can contain alphanumeric, underscore (`_`), and pound (`#`) characters. The `DB_DOMAIN` initialization parameter specifies the database domain name.

The `DB_NAME` parameter and the `DB_DOMAIN` name parameter combine to create the global database name value assigned to the `SERVICE_NAMES` parameter in the initialization parameter file.

The System Identifier (SID) identifies a specific database instance. The SID uniquely distinguishes the instance from any other instance on the same computer. Each database instance requires a unique SID and database name. In most cases, the SID is the same as the database name portion of the global database name.

Locating the Server Parameter File

By default, the preconfigured database uses a server parameter file named `spfilesid.ora`, which is stored in the `$ORACLE_HOME/dbs` directory.

If the server parameter file is not located in the `$ORACLE_HOME/dbs` directory, then the database uses the `SPFILE` parameter in an initialization parameter file to locate it. The default initialization parameter file is `$ORACLE_HOME/dbs/initsid.ora`.

Reviewing Tablespaces and Data Files, Redo Log Files, and Control Files

The following sections contain information about tablespaces and data files, redo log files, and control files:

- [Identifying Tablespaces and Data Files](#)
- [Locating Redo Log Files](#)
- [Locating Control Files](#)

Identifying Tablespaces and Data Files

An Oracle database is divided into smaller logical areas of space known as tablespaces. Each tablespace corresponds to one or more physical data files. Data files contain the contents of logical database structures such as tables and indexes. You can associate each data file with only one tablespace and database.

Note: The `SYSAUX` and `SYSTEM` tablespaces must be present in all Oracle Database 10g databases.

[Table 5–2](#) describes the tablespaces provided by the default preconfigured database.

Table 5–2 *Tablespaces and Descriptions*

Tablespace	Description
EXAMPLE	Stores the sample schemas, if you included them.

Table 5–2 (Cont.) Tablespaces and Descriptions

Tablespace	Description
SYSAUX	Serves as an auxiliary tablespace to the SYSTEM tablespace. Some products and options that previously used the SYSTEM tablespace now use the SYSAUX tablespace, which reduces the load on the SYSTEM tablespace.
SYSTEM	Stores the data dictionary, which includes definitions of tables, views, and stored procedures needed by Oracle Database.
TEMP	Stores temporary tables and indexes when processing SQL statements.
UNDOTBS1	Stores undo information.
USERS	Stores database objects created by database users.

See Also: *Oracle Database Concepts* and the *Oracle Database Administrator's Guide* for more information about tablespaces and data files

Locating Redo Log Files

The preconfigured database uses three redo log files. Redo log files record all changes made to data in the database buffer cache. If an instance fails, then Oracle Database 10g uses the redo log files to recover the modified data in memory.

Oracle Database uses redo log files in a cyclical fashion. For example, if three files constitute the online redo log, Oracle Database fills the first file, then the second file, and then the third file. In the next cycle, it reuses and fills the first file, the second file, and so on.

See Also: *Oracle Database Backup and Recovery Basics* for more information about redo log files

Locating Control Files

The preconfigured database uses three control files. Oracle recommends that you keep at least three control files for each database and set the CONTROL_FILES initialization parameter to specify the location of each file.

A control file is an administrative file. Oracle Database 10g requires a control file to start and run the database. The control file defines the physical structure of the database. For example, it defines the database name and the names and locations of the database data files and redo log files.

Removing Oracle Software

This chapter describes how to completely remove all Oracle databases, instances, and software from an Oracle home directory. It includes information about the following topics:

- [Overview](#)
- [Identifying All Instances](#)
- [Removing an Oracle Database](#)
- [Removing Oracle Software](#)

See Also: The product-specific documentation for requirements and restrictions, to remove an individual product

Overview

To completely remove all Oracle databases, instances, and software from an Oracle home directory, you must:

- Identify all instances associated with the Oracle home.
- Remove database instances.
- Shut down processes.
- Remove the Oracle software.

Identifying All Instances

To identify all instances associated with the Oracle home to be removed, enter the following command:

```
$ more /etc/oratab
```

The output of this command contains entries similar to the following:

```
CUST:/u01/app/oracle/product/10.2.0/db_1:N
```

This entry shows that the CUST Oracle database instance are associated with the /u01/app/oracle/product/10.2.0/db_1 Oracle home directory.

Removing an Oracle Database

To completely remove Oracle Database software, you must remove any installed databases. To remove an Oracle database:

Note: Removing an Oracle database deletes all of the data in the database. To keep this data, ensure that you back up the database before deleting it.

1. Log in as the `oracle` user:

```
$ su - oracle
```
2. Run the `oraenv` or `coraenv` script to set the environment for the database to be removed, for example:
 - For Bourne, Bash, or Korn shell:

```
$ . /usr/local/bin/oraenv
```
 - For C shell:

```
% source /usr/local/bin/coraenv
```
3. At the prompt, specify the SID for the database to be removed.
4. Start the Database Configuration Assistant:

```
$ dbca
```

The Welcome window appears.
5. Click **Next**.
The Operations window appears.
6. Select **Delete a Database**, then click **Next**.
7. Select the database to be deleted, then click **Finish**.
8. In the window that appears, confirm that you want to delete the database.
9. When Database Configuration Assistant removes the database, you are prompted to choose whether you want to perform another operation. Click **Yes** to return to the Operations screen or click **No** to exit from Database Configuration Assistant. To remove another database, click **Yes** and repeat steps 6 through 8.

Removing Oracle Software

The following steps describe how to use Oracle Universal Installer to remove Oracle software from an Oracle home:

Note: Always use Oracle Universal Installer to remove Oracle software. Do not delete any Oracle home directories without first using Oracle Universal Installer to remove the software.

1. If necessary, log in as the `oracle` user:

```
$ su - oracle
```
2. Set the `ORACLE_HOME` environment variable to specify the path of the Oracle home directory to be removed:
 - For Bourne, Bash, or Korn shell:

```
$ ORACLE_HOME=/u01/app/oracle/product/10.2.0/db_1
$ export ORACLE_HOME
```

- For C shell:

```
$ setenv ORACLE_HOME /u01/app/oracle/product/10.2.0/db_1
```

3. Remove any databases instances associated with this Oracle home.

Refer to the preceding sections for information about how to complete these tasks.

4. Stop any processes running in this Oracle home:

Process Name	Command
Oracle Net listener	<code>\$ORACLE_HOME/bin/lsnrctl stop</code>
iSQL*Plus	<code>\$ORACLE_HOME/bin/isqlplusctl stop</code>
Ultra Search	<code>\$ORACLE_HOME/bin/searchctl stop</code>

5. Start Oracle Universal Installer as follows:

```
$ $ORACLE_HOME/oui/bin/runInstaller
```

6. In the Welcome window, click **Deinstall Products**.

The Inventory screen appears, listing all of the Oracle homes on the system.

7. In the Inventory screen, select the Oracle home and the products to be removed, then click **Remove**.

Note: If you choose to remove Oracle JVM, then Oracle Universal Installer removes all installed products that depend on Oracle JVM, including Oracle Database 10g.

Oracle Universal Installer displays a confirmation window asking you to confirm that you want to deinstall the products and their dependent components.

8. Click **Yes**.

Oracle Universal Installer displays a progress indicator as it removes the software.

9. Click **Close** on the Inventory screen.

10. When the products have been deleted, click **Cancel** to exit from Oracle Universal Installer, and then click **Yes**.

Installing and Configuring Oracle Database Using Response Files

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

- [How Response Files Work](#)
- [Creating the oraInst.loc File](#)
- [Preparing a Response File](#)
- [Running Oracle Universal Installer Using a Response File](#)
- [Running NetCA Using a Response File](#)
- [Running Database Configuration Assistant Using a Response File](#)

How Response Files Work

You can automate the installation and configuration of Oracle software, either fully or partially, by specifying a response file when you start Oracle Universal Installer. Oracle Universal Installer uses the values contained in the response file to provide answers to some or all of Oracle Universal Installer prompts.

Typically, Oracle Universal Installer runs in interactive mode, which means that it prompts you to provide information in graphical user interface (GUI) screens. When you use response files to provide this information, you run Oracle Universal Installer at a command prompt using either of the following modes:

- **Silent mode**

If you include responses for all of the prompts in the response file and specify the `-silent` option when starting Oracle Universal Installer, then Oracle Universal Installer runs in silent mode. During a silent-mode installation, Oracle Universal Installer does not display any screens. Instead, it displays progress information in the terminal that you used to start it.
- **Noninteractive (or suppressed) mode**

If you include responses for some or all of the prompts in the response file and omit the `-silent` option, then Oracle Universal Installer runs in suppressed mode. During a suppressed-mode installation, Oracle Universal Installer displays only the screens for which you did not specify all required information. You can also use variables in the response file or command-line options to suppress other installer screens, such as the Welcome screen or Summary screen, that do not prompt for information.

You define the settings for a silent or noninteractive installation by entering values for the variables listed in the response file. For instance, to specify the Oracle home name, you would supply the appropriate value for the `ORACLE_HOME_NAME` variable, as in the following example:

```
ORACLE_HOME_NAME="OraDBHome1"
```

Another way of specifying the variable settings of the response file is to pass them as command line arguments when you run Oracle Universal Installer. For example:

```
$ /directory_path/runInstaller -silent "ORACLE_HOME_NAME=OraDBHome1" ...
```

In this command, *directory_path* is the path of the database directory on the DVD or the path of the `Disk1` directory on the hard drive.

This method is particularly useful to avoid embedding sensitive information, such as passwords, in the response file. For example:

```
$ /directory_path/runInstaller -silent "s_dlgRBOPassword=binks342" ...
```

Ensure that you enclose the variable and its setting in quotes.

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

Reasons for Using Silent Mode or Noninteractive Mode

The following table describes some reasons why you might want to run Oracle Universal Installer in silent mode or suppressed mode.

Mode	Uses
Silent	<p>Use silent mode if you want to:</p> <ul style="list-style-type: none"> ■ Complete an unattended installation, which you might schedule using operating system utilities such as <code>at</code> ■ Complete several similar installations on multiple systems without user interaction ■ Install the software on a system that does not support a GUI <p>Oracle Universal Installer displays progress information in the terminal that you used to start it, but it does not display any of Oracle Universal Installer screens.</p>
Suppressed (noninteractive)	<p>Use suppressed mode if you want to complete similar Oracle software installations on multiple systems, providing default answers to some, but not all of Oracle Universal Installer prompts.</p> <p>If you do not specify information required for a particular Installer screen in the response file, then Oracle Universal Installer displays that screen. It suppresses screens for which you have provided all of the required information.</p>

General Procedure for Using Response Files

The following are the general steps to install and configure Oracle products using Oracle Universal Installer in silent or suppressed mode:

Note: You must complete all required preinstallation tasks on a system before running Oracle Universal Installer in silent or suppressed mode.

1. Create the `oraInst.loc` file.
2. Prepare a response file.
3. Run Oracle Universal Installer in silent or suppressed mode.
4. If you completed a software-only installation, then run NetCA and Database Configuration Assistant in silent or noninteractive mode if required.

These steps are described in the following sections.

Creating the `oraInst.loc` File

If you plan to install Oracle products using Oracle Universal Installer in silent or suppressed mode, then you must manually create the `oraInst.loc` file if it does not exist. This file specifies the location of the Oracle Inventory directory where Oracle Universal Installer creates the inventory of Oracle products installed on the system.

Note: If Oracle software has been installed previously on the system, then the `oraInst.loc` file may exist. The `oraInst.loc` file is present at `/var/opt/oracle/oraInst.loc`. If the file exists, then you must not create a file.

To create the `oraInst.loc` file, follow these steps:

1. Switch user to root:

```
$ su - root
```

2. Change directory as follows:

```
# cd /var/opt/oracle/
```

3. Use a text editor to create the `oraInst.loc` file, containing the following lines:

```
inventory_loc=ORACLE_BASE/oraInventory
inst_group=
```

In this example, `ORACLE_BASE` is the path of the Oracle base directory, for example, `/01/app/oracle`.

4. Enter the following commands to set the appropriate owner, group, and permissions on the `oraInst.loc` file:

```
# chown oracle:oinstall oraInst.loc
# chmod 664 oraInst.loc
```

Preparing a Response File

This section describes the following methods to prepare a response file for use during silent-mode or suppressed-mode installations:

- [Editing a Response File Template](#)
- [Recording a Response File](#)

Editing a Response File Template

This method is most useful for the Enterprise Edition or Standard Edition installation types.

Oracle provides response file templates for each product and installation type, and for each configuration tool. These files are located in the `database/response` directory on the installation media.

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

Table A-1 lists the response files provided with Oracle Database.

Table A-1 Response Files

Response File	Description
<code>enterprise.rsp</code>	Enterprise Edition installation of Oracle Database 10g
<code>standard.rsp</code>	Standard Edition installation of Oracle Database 10g
<code>custom.rsp</code>	Custom installation of Oracle Database 10g
<code>dbca.rsp</code>	Database Configuration Assistant
<code>netca.rsp</code>	Oracle Net Configuration Assistant

To copy and modify a response file:

1. Copy the response file from the response file directory to a directory on your system:

```
$ cp /directory_path/response/response_file.rsp local_directory
```

In this example, `directory_path` is the path to the database directory on the installation media. If you have copied the software to a hard drive, then you can edit the file in the `response` directory if you prefer.

2. Open the response file in a text editor:

```
$ vi /local_dir/response_file.rsp
```

In addition to editing settings specific to the Oracle Database installation, check that the `FROM_LOCATION` path is correct and points to the `products.xml` file in the `stage` directory in the installation media. You can set this variable to point to an absolute path, for example:

```
FROM_LOCATION="/directory_path/stage/products.xml"
```

Remember that you can specify sensitive information, such as passwords, at the command line rather than within the response file. "How Response Files Work" on page A-1 explains this method.

See Also: *Oracle Universal Installer and OPatch User's Guide* for detailed information on creating response files

3. Follow the instructions in the file to edit it.

Note: Oracle Universal Installer or configuration assistant fails if you do not correctly configure the response file. Refer to the "Silent-Mode Response File Error Handling" section on page F-3 for more information about troubleshooting a failed silent-mode installation.

4. Change the permissions on the file to 700:

```
$ chmod 700 /local_dir/response_file.rsp
```

Caution: A fully specified response file for an Oracle Database installation contains the passwords for database administrative accounts and for a user that is a member of the OSDBA group (required for automated backups). Ensure that only the Oracle software owner user can view or modify response files or consider deleting them after the installation succeeds.

Recording a Response File

You can use Oracle Universal Installer in interactive mode to record a response file, which you can edit and then use to complete silent-mode or suppressed-mode installations. This method is useful for custom or software-only installations.

When you record the response file, you can either complete the installation, or you can exit from Oracle Universal Installer on the Summary page, before it starts to copy the software to the system.

If you use record mode during a noninteractive mode 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 during an installation that uses the Basic installation method.

To record a response file:

1. Complete the preinstallation tasks listed in [Chapter 2](#).

When you run Oracle Universal Installer to record a response file, it checks the system to verify that it meets the requirements to install the software. For this reason, Oracle recommends that you complete all of the required preinstallation tasks and record the response file while completing an installation.

2. If you have not installed Oracle software on this system previously, then create the `oraInst.loc` file, as described in the previous section.
3. Ensure that the Oracle software owner user (typically `oracle`) has permissions to create or write to the Oracle home path that you specify when you run Oracle Universal Installer.
4. To record a response file, enter a command similar to the following to start Oracle Universal Installer:

Note: Do not specify a relative path to the response file. If you specify a relative path, then Oracle Universal Installer fails.

```
$ /directory_path/runInstaller -record -destinationFile response_filename
```

In this command:

- `directory_path` is the path of the database directory on the DVD or the path of the `Disk1` directory on the hard drive

- The `-record` parameter specifies that the responses that you enter in a response file must be recorded
 - `response_filename` is the full path and file name of the response file that must be recorded
5. On each Oracle Universal Installer screen, specify the required information.
 6. When Oracle Universal Installer displays the Summary screen, perform one of the following actions:
 - Click **Install** to create the response file, then continue with the installation.
 - Click **Cancel** and then **Yes** to create the response file but exit from Oracle Universal Installer without installing the software.

The response file is saved in the location that you specified using the `-destinationFile` option.

7. If you do not complete the installation, then delete the Oracle home directory that Oracle Universal Installer created using the path you specified on the Specify File Locations screen.
8. Before using the recorded response file on another system, edit the file and make any required changes.

Use the instructions in the file as a guide when editing it.

Running Oracle Universal Installer Using a Response File

Now, you are ready to run Oracle Universal Installer at the command line, specifying the response file you created, to perform the installation. The Oracle Universal Installer executable, `runInstaller`, provides several options. For help information on the full set of these options, run `setup.exe` with the `-help` option, for example:

```
$ directory_path/runInstaller -help
```

The help information appears in a window after some time.

To run Oracle Universal using a response file:

1. Complete the preinstallation tasks listed in [Chapter 2](#).
2. Log in as the Oracle software owner user (typically `oracle`).
3. If you are completing a suppressed-mode installation, then set the `DISPLAY` environment variable.

Note: You do not have to set the `DISPLAY` environment variable if you are completing a silent-mode installation.

4. To start Oracle Universal Installer in silent or suppressed mode, enter a command similar to the following:

Note: Do not specify a relative path to the response file. If you specify a relative path, then Oracle Universal Installer fails.

```
$ /directory_path/runInstaller [-silent] [-noconfig] -responseFile  
responsefilename
```

In this example:

- *directory_path* is the path of the database directory on the DVD or the path of the `Disk1` directory on the hard drive.
- `-silent` indicates that Oracle Universal Installer must be run in silent mode.
- `-noconfig` suppresses running the configuration assistants during installation, and a software-only installation is performed instead.
- *responsefilename* is the full path and file name of the installation response file that you configured.

Note: For more information about other options for the `runInstaller` command, enter the following command:

```
$ /directory_path/runInstaller -help
```

5. When the installation completes, log in as the `root` user and run the `root.sh` script:

```
$ su
# /oracle_home_path/root.sh
```

Running NetCA Using a Response File

You can run NetCA in silent mode to configure and start an Oracle Net listener on the system, configure naming methods, and configure Oracle Net service names. To run NetCA in silent mode, you must copy and edit a response file template. Oracle provides a response file template named `netca.rsp` in the `response` directory in the `database/response` directory on the DVD.

Note: If you copied the software to a hard disk, then the response file template is located in the `Disk1/response` directory.

To run NetCA using a response file:

1. Copy the `netca.rsp` response file template from the response file directory to a directory on your system:

```
$ cp /directory_path/response/netca.rsp local_directory
```

In this example, *directory_path* is the path of the database directory on the DVD. If you have copied the software to a hard drive, then you can edit the file in the `response` directory if you prefer.

2. Open the response file in a text editor:

```
$ vi /local_dir/netca.rsp
```

3. Follow the instructions in the file to edit it.

Note: NetCA fails if you do not correctly configure the response file.

4. Log in as the Oracle software owner user, and set the `ORACLE_HOME` environment variable to specify the correct Oracle home directory.

5. Enter a command similar to the following to run NetCA in silent mode:

```
$ $ORACLE_HOME/bin/netca /silent /responsefile /local_dir/netca.rsp
```

In this command:

- The `/silent` option indicates that NetCA must be run in silent mode.
- `local_dir` is the full path of the directory where you copied the `netca.rsp` response file template.

Running Database Configuration Assistant Using a Response File

You can run Database Configuration Assistant in noninteractive or silent mode to configure and start an Oracle Database on your system. To run Database Configuration Assistant in noninteractive or silent mode, you must copy and edit a response file template. Oracle provides a response file template named `dbca.rsp` in the `database/response` directory on the DVD.

Note: If you copied the software to a hard disk, then the response file template is located in the `Disk1/response` directory.

This section contains the following topics:

- [Using Database Configuration Assistant in Noninteractive Mode](#)
- [Using Database Configuration Assistant in Silent Mode](#)
- [Running Database Configuration Assistant in Noninteractive or Silent Mode](#)

Using Database Configuration Assistant in Noninteractive Mode

In noninteractive mode, Database Configuration Assistant uses values that you specify, in the response file or as command line options, to create a database. As it configures and starts the database, it displays a window that contains status messages and a progress bar. The window that it displays is the same window that is displayed when you choose to create a preconfigured database during an Enterprise Edition or Standard Edition installation.

To run Database Configuration Assistant in noninteractive mode, you must use a graphical display and set the `DISPLAY` environment variable.

Using Database Configuration Assistant in Silent Mode

In silent mode, Database Configuration Assistant uses values that you specify, in the response file or as command line options, to create a database. Database Configuration Assistant does not display any screens or information while it creates the database.

Running Database Configuration Assistant in Noninteractive or Silent Mode

To run Database Configuration Assistant in noninteractive or silent mode:

Note: As an alternative to editing the response file template, you can also create a database by specifying all required information as command line options when you run Database Configuration Assistant. For information about the list of options supported, enter the following command:

```
$ $ORACLE_HOME/bin/dbca -help
```

1. Copy the `dbca.rsp` response file template from the response file directory to a directory on your system:

```
$ cp /directory_path/response/dbca.rsp local_directory
```

In this example, `directory_path` is the path of the database directory on the DVD. If you have copied the software to a hard drive, then you can edit the file in the response directory if you prefer.

2. Open the response file in a text editor:

```
$ vi /local_dir/dbca.rsp
```

3. Edit the file, following the instructions in the file.

Note: Database Configuration Assistant fails if you do not correctly configure the response file.

4. Log in as the Oracle software owner user, and set the `ORACLE_HOME` environment variable to specify the correct Oracle home directory.
5. If you intend running Database Configuration Assistant in noninteractive mode, then set the `DISPLAY` environment variable.
6. Enter a command similar to the following to run Database Configuration Assistant in noninteractive or silent mode with a response file:

```
$ $ORACLE_HOME/bin/dbca {-progressOnly | -silent} [-cloneTemplate] \  
[-datafileDestination /datafilepath] -responseFile /local_dir/dbca.rsp
```

In this example:

- The `-silent` option indicates that Database Configuration Assistant must be run in silent mode.
- The `-progressOnly` option indicates that Database Configuration Assistant must be run in noninteractive mode.
- The `-cloneTemplate` option is required if one of the following preconfigured databases must be created:
 - General Purpose
 - Transaction Processing
 - Data Warehouse
- `-datafileDestination` specifies the parent directory where the database files must be created. If you do not specify this option, then the parent directory for database files is the `$ORACLE_BASE/oradata` directory if the `ORACLE_BASE` environment variable is set, or the `$ORACLE_HOME/oradata` directory if `ORACLE_BASE` is not set.

- *local_dir* is the full path of the directory where you copied the `dbca.rsp` response file template.

Cloning an Oracle Home

Cloning an Oracle home involves creating a copy of the Oracle home and then configuring it for a new environment. If you perform multiple Oracle Database installations, then you may use this method to create each Oracle home, because copying files from an existing Oracle Database installation takes less time than creating a new version of them. This method is also useful if the Oracle home that is to be cloned has had patches applied to it. When you clone this Oracle home, the new Oracle home has the patch updates as well.

To clone an Oracle home:

1. Verify that the installation of Oracle Database to be cloned is successful.

You can do this by reviewing the `installActionsdate_time.log` file for the installation session, which is normally located in the `/oraInventory_location/logs` directory.

If you installed patches, then you can check their status by running the following commands:

```
$ $ORACLE_HOME/OPatch ORACLE_HOME=ORACLE_HOME_using_patch
$ $ORACLE_HOME/OPatch opatch lsinventory
```

2. Stop all processes related to the Oracle home. Refer to the "[Removing Oracle Software](#)" section on page 6-2 for more information on stopping the processes for an Oracle home.
3. Create a ZIP file with the Oracle home (but not Oracle base) directory.

For example, if the source Oracle installation is in the `/u01/app/oracle/product/10.2.0/db_1` directory, then zip the `db_1` directory by using the following command:

```
# zip -r db_1.zip /u01/app/oracle/product/10.2.0/db_1
```

Leave out the `admin`, `flash_recovery_area`, and `oradata` directories that are in the `10.2.0` directory. These directories are created in the target installation, when you create a new database.

4. Copy the ZIP file to the root directory of the target computer.
5. Extract the ZIP file contents by using the following command:

```
# unzip -d / db_1.zip
```

6. Repeat steps 4 and 5 for each computer where the Oracle home must be cloned, unless the Oracle home is on a shared storage device.

-
7. On the target computer, change directory to the unzipped Oracle home directory, and remove all the .ora (*.ora) files present in the unzipped \$ORACLE_HOME/network/admin directory.

8. From the \$ORACLE_HOME/oui/bin directory, run Oracle Universal Installer in clone mode for the unzipped Oracle home. Use the following syntax:

```
$ORACLE_HOME/oui/bin/runInstaller -silent -clone ORACLE_HOME="target location"  
ORACLE_HOME_NAME="unique_name_on_node" [-responseFile full_directory_path]
```

For example:

```
$ORACLE_HOME/oui/bin/runInstaller -silent -clone ORACLE_  
HOME="/u01/app/oracle/product/10.2.0/db_1" ORACLE_HOME_NAME="db_1"
```

The `-responseFile` parameter is optional. You can supply clone-time parameters on the command line or by using the response file named on the command line.

Oracle Universal Installer starts, and then records the cloning actions in the `cloneActionstimestamp.log` file. This log file is normally located in the `/orainventory_location/logs` directory.

9. To create a new database for the cloned Oracle home, run Database Configuration Assistant as follows:

```
$ cd $ORACLE_HOME/bin  
$ ./dbca
```

10. To configure connection information for the new database, run Net Configuration Assistant.

```
$ cd $ORACLE_HOME/bin  
$ ./netca
```

See Also:

- *Oracle Universal Installer and OPatch User's Guide* for detailed information about using Oracle Universal Installer to clone an Oracle Database home
- *Oracle Database Administrator's Guide* for information about cloning an Oracle database, and also, cloning an Oracle Database home

Optimal Flexible Architecture

This appendix describes the Optimal Flexible Architecture standard. The standard is a set of configuration guidelines created to ensure reliable Oracle installations that require little maintenance. It includes information about the following topics:

- [Overview of the Optimal Flexible Architecture Standard](#)
- [Changes to the Optimal Flexible Architecture for Oracle Database 10g](#)
- [Implementing Optimal Flexible Architecture](#)
- [Improving Reliability and Performance](#)

Overview of the Optimal Flexible Architecture Standard

The Optimal Flexible Architecture standard is designed to:

- Organize large amounts of complicated software and data on disk, to avoid device bottlenecks and poor performance
- Facilitate routine administrative tasks such as software and data backup, which are often vulnerable to data corruption
- Facilitate switching between multiple Oracle databases
- Adequately manage and administer database growth
- Help eliminate fragmentation of free space in the data dictionary, isolate other fragmentation, and minimize resource contention

Optimal Flexible Architecture is a set of guidelines that you must adopt when organizing Oracle directories and files on your computer. All Oracle components on the installation media are compliant with Optimal Flexible Architecture. That is, Oracle Universal Installer places Oracle Database components in directory locations that follow Optimal Flexible Architecture guidelines.

Although using Optimal Flexible Architecture is not a requirement, Oracle recommends that you use it if your database will grow, or if you plan to have multiple databases.

Characteristics of an Optimal Flexible Architecture Compliant Installation

The following are the characteristics of an Oracle product installation that complies with the Optimal Flexible Architecture standard:

- File system organization

The file system is organized to enable easy administration and to facilitate:

- Adding data into existing databases
- Adding users
- Creating databases
- Adding hardware
- Distributed I/O loads
I/O loads are distributed across enough disk drives to prevent performance bottlenecks.
- Hardware support
In most cases, you do not require new hardware to implement the Optimal Flexible Architecture standard.
- Safeguards Against Drive Failures
By distributing applications across multiple drives, drive failures affect as few applications as possible.
- Distribution of Oracle home directories
The following items can be distributed across multiple disk drives:
 - The collection of home directories
 - The contents of an individual home directory
- Integrity of login home directories
You can add, move, or delete login home directories without having to revise programs that refer to them.
- Independence of UNIX directory subtrees
Categories of files are separated into independent UNIX directory subtrees so that files in one category are minimally affected by operations on files in other categories.
- Supports concurrent execution of application software
You can run multiple versions of Oracle software simultaneously, enabling you to test and use a new release before retiring the previous release. Transferring to a new release after an upgrade is simple for the administrator and transparent for the user.
- Separates administrative information for each database
The ability to separate administrative information for each database ensures a reasonable structure for the organization and storage of administrative data.
- Uses consistent database file naming
Database files are named so that:
 - Database files are easy to distinguish from other files.
 - Files belonging to one database are easy to distinguish from files that belong to another database.
 - Control files, redo log files, and data files can be identified as such.
 - The association of data file to tablespace is clearly indicated.
- Separation of tablespace contents
Tablespace contents are separated to:

- Minimize tablespace free space fragmentation.
- Minimize I/O request contention.
- Maximize administrative flexibility.
- I/O loads tuned across all drives
I/O loads are tuned across all drives.

Changes to the Optimal Flexible Architecture for Oracle Database 10g

For previous releases of Oracle Database, the Optimal Flexible Architecture standard recommended Oracle home path was similar to the following:

```
/u01/app/oracle/product/9.2.0
```

For Oracle Database 10g, the Optimal Flexible Architecture recommended Oracle home path has changed. The Optimal Flexible Architecture recommended path is now similar to the following:

```
/u01/app/oracle/product/10.2.0/type[_n]
```

In this example, *type* is the type of Oracle home, for example Oracle Database (*db*) or Oracle Database Client (*client*), and *n* is an optional counter. This syntax provides the following benefits:

- You can install different products with the same release number in the same Oracle base directory, for example:

```
/u01/app/oracle/product/10.2.0/db_1  
/u01/app/oracle/product/10.2.0/client_1
```

- You can install the same product more than once in the same Oracle base directory, for example:

```
/u01/app/oracle/product/10.2.0/db_1  
/u01/app/oracle/product/10.2.0/db_2
```

Implementing Optimal Flexible Architecture

This section describes the naming strategy recommended by the Optimal Flexible Architecture standard. It contains the following sections:

- [File Systems](#)
- [Naming Directories](#)
- [Naming Database Files](#)
- [Separating Segments with Different Requirements](#)
- [Naming Tablespaces](#)
- [Exploiting the Optimal Flexible Architecture Structure for Oracle Files](#)
- [Optimal Flexible Architecture File Mapping](#)

File Systems

This section describes the conventions for mount points.

Number of File Systems

To fully implement the Optimal Flexible Architecture recommendations for a database stored on file systems that are not striped or mirrored, you require at least three file systems located on separate physical devices.

Naming Conventions

Name all file system mount points using the syntax `/pm`, where `p` is a string constant and `m` is a unique fixed-length key (typically a two-digit number) used to distinguish each mount point. For example: `/u01` and `/u02`, or `/disk01` and `/disk02`.

Naming Mount Points for Very Large Databases (VLDBs)

If each disk drive contains database files from one application and there are enough drives for each database to prevent I/O bottlenecks, then use the syntax `/pm/q/dm` for naming mount points. [Table C-1](#) describes the variables used in this syntax.

Table C-1 Syntax for Naming Mount Points for Very Large Databases

Variable	Description
<code>pm</code>	A mount point name
<code>q</code>	A string denoting that Oracle data is stored in this directory, for example, <code>oradata</code>
<code>dm</code>	The value of the initialization parameter <code>DB_NAME</code> (typically the same as the instance SID for single-instance databases)

For example, to allocate two drives exclusively for the `test` database, name the mount points `/u01/oradata/test` and `/u02/oradata/test`.

Naming Directories

This section describes the naming conventions for directories that are compliant with the Optimal Flexible Architecture standard.

Oracle Base Directory Naming Convention

The Oracle base directory is the top-level directory for Oracle products installed by the same user. Name Oracle base directories using the syntax `/pm/h/u`. [Table C-2](#) describes the variables used in this syntax.

Table C-2 Syntax for Naming Oracle Base Directories

Variable	Description
<code>pm</code>	A mount point name
<code>h</code>	A standard directory name
<code>u</code>	The name of the owner of the directory (the user running Oracle Universal Installer)

For example, `/u01/app/oracle` is an Oracle base directory created by the `oracle` user and `/u01/app/applmgr` is an Oracle base directory created by the `applmgr` user.

Placing Oracle base directories at the same level in the UNIX file system is advantageous because it enables you to refer to the collection of Oracle base directories on different mount points using a single pattern matching string, `/*/app/*`.

Referring to Path Names

Refer to explicit path names only in files designed specifically to store them, such as the password file, `/etc/passwd`, and the Oracle `oratab` file. Refer to group memberships only in the `/etc/group` file.

Oracle Home Directory Naming Convention

To help fulfill the Optimal Flexible Architecture requirement of simultaneously running multiple versions of Oracle software, install the software in a directory matching the pattern `/pm/h/u/product/v/type_[n]`.

Table C-3 describes the variables used in this syntax.

Table C-3 Syntax for Naming Oracle Home Directories

Variable	Description
<i>pm</i>	A mount point name
<i>h</i>	A standard directory name
<i>u</i>	The name of the owner of the directory
<i>v</i>	The version of the software
<i>type</i>	The type of installation, for example Database (<code>db</code>), Client (<code>client</code>), Companion (<code>companion</code>), or CRS (<code>crs</code>)
<i>n</i>	An optional counter, which enables you to install the same product more than once in the same Oracle base directory

For example:

- `/u01/app/oracle/product/10.2.0/db_1` indicates the Oracle home directory for the first installation of Oracle Database on this system.

Set the `ORACLE_HOME` environment variable after installation to specify the Oracle home directory.

Naming Subdirectories

To facilitate the organization of administrative data, Oracle recommends that you store database-specific administration files in subdirectories matching the pattern `/h/admin/d/a/`, where *h* is the Oracle base directory, *d* is the database name (DB_NAME), and *a* is a subdirectory for specific types of database administration files.

Table C-4 describes the database administration file subdirectories.

Table C-4 Subdirectories for Database Administration Files

Subdirectory	Description
<code>adhoc</code>	Ad hoc SQL scripts
<code>arch</code>	Archived redo log files
<code>adump</code>	Audit files (Set the <code>AUDIT_FILE_DEST</code> initialization parameter to specify the <code>adump</code> directory. Clean out this subdirectory periodically.)
<code>bdump</code>	Background process trace files
<code>cdump</code>	Core dump files
<code>create</code>	Scripts used to create the database
<code>exp</code>	Database export files
<code>logbook</code>	Files recording the status and history of the database

Table C-4 (Cont.) Subdirectories for Database Administration Files

Subdirectory	Description
pfile	Instance parameter files
udump	User SQL trace files

For example, `/u01/app/oracle/admin/sab/adhoc/` is the `adhoc` subdirectory associated with the database named `sab`.

Naming Database Files

The following table lists the recommended file naming conventions for database files:

Note: Oracle Managed Files (OMF) use different naming conventions. For more information about these naming conventions, refer to the *Oracle Database Administrator's Guide*.

File Type	File Naming Convention
Control files	<code>/pm/q/d/control.ctl</code>
Redo log files	<code>/pm/q/d/redon.log</code>
Data files	<code>/pm/q/d/tn.dbf</code>

The following table describes this syntax:

Variable	Description
<i>pm</i>	A mount point name described previously in this appendix
<i>q</i>	A string (typically <code>oradata</code>) distinguishing Oracle data from all other files
<i>d</i>	The value of the <code>DB_NAME</code> initialization parameter (typically, the same as the instance <code>SID</code> for single-instance databases)
<i>t</i>	An Oracle tablespace name
<i>n</i>	A two-digit string

Note: Do not store files other than control files, redo log files, or data files associated with database *d* in the path `/pm/q/d`.

Using this convention, it is easy to determine the database to which the `/u03/oradata/sab/system01.dbf` file belongs.

Separating Segments with Different Requirements

Separate groups of segments with different life spans, I/O request demands, and backup frequencies across different tablespaces.

Table C-5 describes the special tablespaces that the Database Configuration Assistant creates for each Oracle database. If you manually create a database, then you must create the required tablespaces. These tablespaces are in addition to those required for application segments.

See Also: *Oracle Database Administrator's Guide* for information about creating databases manually

Table C-5 *Special Tablespaces*

Tablespace	Required	Description
EXAMPLE	No	The EXAMPLE tablespace used to store the Sample Schemas
SYSAUX	Yes	Auxiliary tablespace to the SYSTEM tablespace
SYSTEM	Yes	Data dictionary segments
TEMP	Yes	Temporary segments
UNDOTBS1	Yes	Used by Oracle to store undo information
USERS	No	Miscellaneous user segments

Creating these special tablespaces is effective because data dictionary segments are never dropped, and no other segments that can be dropped are allowed in the SYSTEM tablespace. Doing this ensures that the SYSTEM tablespace does not require a rebuild due to tablespace free-space fragmentation.

Naming Tablespaces

Name tablespaces descriptively using a maximum of eight characters. Although Oracle Database tablespace names can be 30 characters long, portable UNIX file names are restricted to 14 characters. The recommended standard for a data file base name is *tn.dbf*, where *t* is a descriptive tablespace name and *n* is a two-digit string. Because the extension and the two-digit string take six characters, only eight characters remain for the tablespace name.

Descriptive names enable the data file to be associated with the tablespace that uses it. For example, the names GLD and GLX might be used for the tablespaces storing General Ledger data and General Ledger indexes, respectively.

Note: Do not embed reminders of the word "tablespace" in your tablespace names. Tablespace names can be distinguished by context. For example, do not name the General Ledger tablespace `GLD_TBS01.dbf`.

Exploiting the Optimal Flexible Architecture Structure for Oracle Files

Table C-6 describes the syntax used for identifying classes of files.

Table C-6 *Directory Structure Syntax for Identifying Classes of Files*

Directory Structure Syntax	Description
<code>/u[0-9][0-9]</code>	User data directories
<code>*/home/*</code>	User home directories
<code>*/app/*</code>	User application software directories
<code>*/app/applmgr</code>	Oracle applications software subtrees
<code>*/app/oracle/product</code>	Oracle software subtrees
<code>*/app/oracle/product/10.2.0</code>	Oracle software subtree for release 10.2.0 products

Table C-6 (Cont.) Directory Structure Syntax for Identifying Classes of Files

Directory Structure Syntax	Description
<code>/*/app/oracle/product/10.2.0/db*</code>	Oracle home directories for Oracle Database 10g
<code>/*/app/oracle/admin/sab</code>	sab database administrative subtrees
<code>/*/app/oracle/admin/sab/arch/*</code>	sab database archived log files
<code>/*/oradata</code>	Oracle data directories
<code>/*/oradata/sab/*</code>	sab database files
<code>/*/oradata/sab/*.log</code>	sab database redo log files

Optimal Flexible Architecture File Mapping

Table C-7 shows a hierarchical file mapping of a sample Optimal Flexible Architecture-compliant installation with two Oracle home directories and two databases. The database files are distributed across three mount points, `/u02`, `/u03`, and `/u04`.

Table C-7 Hierarchical File Mapping for an Optimal Flexible Architecture Installation

Directory	Description
<code>/</code>	Root directory
<code>/u01/</code>	User data mount point 1
<code>/u01/app/</code>	Subtree for application software
<code>/u01/app/oracle/</code>	Oracle Base directory
<code>/u01/app/oracle/admin/</code>	Subtree for database administration files
<code>/u01/app/oracle/admin/TAR</code>	Subtree for support log files
<code>/u01/app/oracle/admin/db_name1/</code>	admin subtree for <i>db_name1</i> database
<code>/u01/app/oracle/admin/db_name2/</code>	admin subtree for <i>db_name2</i> database
<code>/u01/app/oracle/doc/</code>	Online documentation
<code>/u01/app/oracle/flash_recovery_area/</code>	Subtree for recovery files
<code>/u01/app/oracle/flash_recovery_area/db_name1</code>	Recovery files for <i>db_name1</i> database
<code>/u01/app/oracle/flash_recovery_area/db_name2</code>	Recovery files for <i>db_name2</i> database
<code>/u01/app/oracle/product/</code>	Distribution files
<code>/u01/app/oracle/product/9.2.0</code>	Oracle home directory for Oracle9i release 2 (9.2.0)
<code>/u01/app/oracle/product/10.2.0/db_1</code>	Oracle home directory for Oracle Database 10g release 2 (10.2)
<code>/u01/app/kjff/</code>	Oracle base directory for user <i>kjff</i>
<code>/u01/app/edmf/</code>	Oracle base directory for user <i>edmf</i>
<code>/u02</code>	User data mount point 2
<code>/u02/oradata/</code>	Subtree for Oracle data
<code>/u02/oradata/db_name1/</code>	Subtree for <i>db_name1</i> database files
<code>/u02/oradata/db_name2/</code>	Subtree for <i>db_name2</i> database files
<code>/u03/</code>	User data mount point 3
<code>/u03/oradata/</code>	Subtree for Oracle data

Table C-7 (Cont.) Hierarchical File Mapping for an Optimal Flexible Architecture Installation

Directory	Description
/u03/oradata/db_name1/	Subtree for <i>db_name1</i> database files
/u03/oradata/db_name2/	Subtree for <i>db_name2</i> database files
/u04/	User data mount point 4
/u04/oradata/	Subtree for Oracle data
/u04/oradata/db_name1/	Subtree for <i>db_name1</i> database files
/u04/oradata/db_name2/	Subtree for <i>db_name2</i> database files

Improving Reliability and Performance

One of the goals of Optimal Flexible Architecture is to improve reliability and performance by distributing I/O loads across different physical drives. The following are methods to accomplish this:

- [Disk Mirroring](#)
- [Disk Striping](#)

Disk Mirroring

You can separate and treat Oracle Database log files and database files with different levels of hardware reliability. Oracle Database log files are highly reliable to start with, because they are stored redundantly. Creating similar reliability for database files may require you to duplicate all of your data, using disk mirrors.

Disk mirroring usually involves two or more identical drives and a RAID controller. If one disk fails, then the other disks can recover data that would otherwise be lost. Using one of the disks to recover lost data may involve losing the mirror. If this happens, then you must build a new mirror.

Disk mirroring is part of some levels of Redundant Array of Independent Disks (RAID) configurations, provided by the disk controller. The RAID level determines the amount of redundancy. Some RAID levels can use the hot swapping feature, which means that you can replace a bad disk with a good one without turning off the computer or losing functionality.

Disk Striping

How you set up disks for use in a database depends on the number of disks and the type of hard disk controllers available. If the hard disk controllers support both striping and mirroring, then Oracle recommends that you configure the controllers to support striping.

Striping provides significant performance advantages. All the space from the striped drives appears as a single logical drive. In addition, the space is used by interlacing stripes of space from all of the disks in the stripe. That is, a large file uses some space from the first disk, then some from the second disk, and so on to the last disk, and then starting back at the first disk again. Each file can be spread over all of the striped disks. Multiple CPUs can access data randomly in such a file without contention.

Controllers that support striping usually provide caching. That is, data can be written to the controller and cached and saved for a time in storage not on the disk. Data that is read can be cached on the controller in the same manner. Read caching must not be used with Oracle Database, because all database reads are already cached in the

System Global Area (SGA). The value of the `DB_CACHE_SIZE` parameter in the initialization parameter file, `init.ora`, determines the buffer size that can be used in the SGA. This value also configures Oracle Database on startup.

Note:

- Read caching must be disabled.
 - Disk write caching must be disabled on disks containing Oracle Database data files and redo log files where the contents of the write cache are not flushed to disk if a power failure or operating system failure occurs. Consult your vendor documentation for more information.
-
-

Managing Oracle Database Port Numbers

During installation, Oracle Universal Installer assigns port numbers to components from a set of default port numbers. This appendix lists the default port numbers and describes how to change the assigned port after installation. It includes information about the following topics:

- [About Managing Ports](#)
- [Viewing Port Numbers and Access URLs](#)
- [Port Numbers and Protocols of Oracle Components](#)
- [Changing the iSQL*Plus Ports](#)
- [Changing the Oracle Ultra Search Ports](#)
- [Changing the Oracle XML DB Ports](#)

About Managing Ports

During installation, Oracle Universal Installer assigns port numbers to components from a set of default port numbers. Many Oracle Database components and services use ports. As an administrator, it is important to know the port numbers used by these services, and to ensure that the same port number is not used by two services on your host.

Most port numbers are assigned during installation. Every component and service has an allotted port range, which is the set of port numbers Oracle Database attempts to use when assigning a port. Oracle Database starts with the lowest number in the range and performs the following checks:

- Is the port used by another Oracle Database installation on the host?
The installation may be up or down at the time; Oracle Database can still detect if the port is used.
- Is the port used by a process that is currently running?
This could be any process on the host, even a non-Oracle Database process.
- Is the port listed in the `/etc/services` files?

If the answer to any of the preceding questions is yes, then Oracle Database moves to the next highest port in the allotted port range and continues checking until it finds a free port.

Viewing Port Numbers and Access URLs

In most cases, the Oracle Database component's port number is listed in the tool used to configure the port. In addition, ports for some Oracle Database applications are listed in the `portlist.ini` file. This file is located in the `$ORACLE_HOME/install` directory.

If you change a port number, it is not updated in the `portlist.ini` file, so you can only rely on this file immediately after installation. To find or change a port number, use the methods described in this appendix.

Port Numbers and Protocols of Oracle Components

The following table lists the port numbers and protocols used by components that are configured during the installation. By default, the first port in the range is assigned to the component, if it is available.

Table D-1 Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol
Oracle SQL*Net Listener Allows Oracle client connections to the database over Oracle's SQL*Net protocol. You can configure it during installation. To reconfigure this port, use Net Configuration Assistant.	1521	1521	TCP
Data Guard Shares the SQL*Net port and is configured during installation. To reconfigure this port, use Net Configuration Assistant to reconfigure the Oracle SQL*Net listener.	1521 (same value as the listener)	1521	TCP
Connection Manager Listening port for Oracle client connections to Oracle Connection Manager. It is not configured during installation, but can be configured using Net Configuration Assistant.	1630	1630	TCP
iSQL*Plus HTTP port for iSQL*Plus. The port number is automatically assigned during installation. "Changing the iSQL*Plus Ports" on page D-3 explains how to change its port number.	5560	5560–5579	TCP/HTTP
iSQL*Plus RMI port for iSQL*Plus. The port number is automatically assigned during installation. "Changing the iSQL*Plus Ports" on page D-3 explains how to change its port number.	5580	5580–5599	TCP
iSQL*Plus JMS port for iSQL*Plus. The port number is automatically assigned during installation. "Changing the iSQL*Plus Ports" on page D-3 explains how to change its port number.	5600	5600–5619	TCP

Table D-1 (Cont.) Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol
Oracle Ultra Search HTTP port for Oracle Ultra Search. Its port number is assigned automatically when you install Oracle Ultra Search, by using the Custom installation type. " Changing the Oracle Ultra Search Ports " on page D-4 explains how to change its port number.	5620	5620–5639	TCP/HTTP
Oracle Ultra Search RMI port for Oracle Ultra Search. Its port number is assigned automatically when you install Oracle Ultra Search, by using the Custom installation type. " Changing the Oracle Ultra Search Ports " on page D-4 explains how to change its port number.	5640	5640–5659	TCP
Oracle Ultra Search JMS port for Oracle Ultra Search. Its port number is assigned automatically when you install Oracle Ultra Search, by using the Custom installation type. " Changing the Oracle Ultra Search Ports " on page D-4 explains how to change its port number.	5660	5660–5679	TCP
Oracle XML DB The Oracle XML DB HTTP port is used if Web-based applications must access an Oracle database from an HTTP listener. It is configured during installation, but you cannot view it afterward. " Changing the Oracle XML DB Ports " on page D-4 explains how to change its port number.	Dynamic	Dynamic	HTTP
Oracle XML DB The Oracle XML DB FTP is used when applications must access an Oracle database from an FTP listener. It is configured during installation, but you cannot view it afterward. " Changing the Oracle XML DB Ports " on page D-4 explains how to change its port number.	Dynamic	Dynamic	FTP
Oracle Event Manager The port number is assigned automatically during installation. You cannot view or modify it afterward.	49897	49897–49898	TCP

Changing the iSQL*Plus Ports

The following sections describe how to change the iSQL*Plus ports.

Changing the HTTP Port

To change the HTTP port, edit the following files:

- `$ORACLE_HOME/host_sid/sysman/config/emoms.properties`

Modify the following port parameters (for example, 5560) in the file:

```
oracle.sysman.db.isqlplusUrl=http\://host.domain\:5560/isqlplus/dynamic
oracle.sysman.db.isqlplusWebDBAUrl=http\://host.domain\:5560/isqlplus/dynamic
```

- `$ORACLE_HOME/oc4j/j2ee/isqlplus/config/http-web-site.xml`

Modify the port attribute of the web-site element:

```
<web-site port="5560" ...>
```

Changing the RMI Port

To change the RMI port, modify the `port` attribute of the `rmi-server` element in the `$ORACLE_HOME/oc4j/j2ee/isqlplus/config/rmi.xml` file:

```
<rmi-server port="5580"...>
```

Changing the JMS Port

To change the JMS port, modify the `port` attribute of the `jms-server` element in the `$ORACLE_HOME/oc4j/j2ee/isqlplus/config/jms.xml` file:

```
<jms-server port="5600"...>
```

Changing the Oracle Ultra Search Ports

The following sections describe how to change the Oracle Ultra Search ports.

Changing the HTTP Port

To change the HTTP port, modify the `port` attribute of the `web-site` element in the `$ORACLE_HOME/oc4j/j2ee/OC4J_SEARCH/config/http-web-site.xml` file:

```
<web-site port="5620"...>
```

Changing the RMI Port

To change the RMI port, modify the `port` attribute of the `rmi-server` element in the `$ORACLE_HOME/oc4j/j2ee/OC4J_SEARCH/config/rmi.xml` file:

```
<rmi-server port="5640"...>
```

Changing the JMS Port

To change the JMS port, modify the `port` attribute of the `jms-server` element in the `$ORACLE_HOME/oc4j/j2ee/OC4J_SEARCH/config/jms.xml` file:

```
<jms-server port="5660"...>
```

Changing the Oracle XML DB Ports

To change the Oracle XML DB FTP and HTTP ports, you must run the `catxdbdbca.sql` script, which in a default installation is located in `$ORACLE_HOME/rdbms/admin`.

To change the Oracle XML DB ports:

1. Check that the Oracle listener is running. To do so, in the Services control panel, ensure that the Oracle TNS Listener service (for example, `OracleOraDb10g_home1TNSListener`) is set to **Started**.

If you cannot start the listener, refer to *Oracle Database Net Services Administrator's Guide*.

2. Log into SQL*Plus or iSQL*Plus as SYS or XDB using the SYSDBA role.

For example, to log in to SQL*Plus as SYS:

```
# sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

3. Run the `catxdbdbca.sql` script.

For example, to use 2200 for the FTP port and 8200 for the HTTP port, and assuming your Oracle home is in the following location, enter the following command:

```
SQL> $ORACLE_HOME/rdbms/admin/catxdbdbca.sql 2200 8200
```

4. Exit SQL*Plus or iSQL*Plus.

Configuring Oracle Database Globalization Support

This appendix describes the following Globalization Support topics:

- [About NLS_LANG Environment Variable](#)
- [Running Oracle Universal Installer in Different Languages](#)

About NLS_LANG Environment Variable

Oracle provides Globalization Support that enables users to interact with a database in their preferred locale and character set settings. Setting the NLS_LANG environment variable specifies locale behavior for Oracle software. It sets the language and territory used by the client application and the database server. It also sets the character set for entering and displaying data by a client program, such as SQL*Plus.

The NLS_LANG environment variable uses the following format:

`NLS_LANG=language_territory.characterset`

where:

Parameter	Description
<i>language</i>	Specifies the language used for displaying Oracle messages, sorting, day names, and month names.
<i>territory</i>	Specifies the conventions for default date, monetary and numeric formats.
<i>characterset</i>	Specifies the encoding used by the client application (normally the Oracle character set that corresponds to the character set of the user terminal or the operating system)

The NLS_LANG environment variable is set as a local environment variable for the shell on all UNIX-based platforms. For example, if the operating system locale setting is `en_US.UTF-8`, then the corresponding NLS_LANG environment variable must be set to `AMERICAN_AMERICA.AL32UTF8`.

See Also: *Oracle Database Globalization Support Guide* for information about the NLS_LANG parameter and Globalization Support initialization parameters

The following table lists some valid values for the NLS_LANG environment variable.

Note: Refer to the operating system specific documentation on how to determine the operating system locale environment setting.

Operating system locale	NLS_LANG values
French (France)	FRENCH_FRANCE.WE8ISO8859P15, FRENCH_FRANCE.WE8ISO8859P1, FRENCH_FRANCE.WE8MSWIN1252, FRENCH_FRANCE.AL32UTF8
Japanese (Japan)	JAPANESE_JAPAN.JA16EUC, JAPANESE_JAPAN.JA16SJIS, JAPANESE_JAPAN.AL32UTF8

Running Oracle Universal Installer in Different Languages

Oracle Universal Installer runs by default in the selected language of your operating system. You can also run Oracle Universal Installer in the following languages:

- Brazilian Portuguese
- German
- Japanese
- Simplified Chinese
- Traditional Chinese
- French
- Italian
- Korean
- Spanish

To run Oracle Universal Installer in a different language, you can use any of the following methods:

- Change the language in which the operating system is running before you run Oracle Universal Installer. You can change the language in which the operating system is running by clicking **Language** on the Login screen and selecting the required language from the list.
- To run Oracle Universal Installer in another language from the shell prompt, use a command similar to the following after changing to the Disk1 directory (or to the directory that contains the `runInstaller` file):

```
# LANG=language_territory.characterset ./runInstaller
```

For example, to run Oracle Universal Installer in German, use the following command:

```
# LANG=de_DE.UTF-8 ./runInstaller
```

See Also: *Oracle Universal Installer and OPatch User's Guide* for information about running Oracle Universal Installer in different languages

Troubleshooting

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

- [Verify Requirements](#)
- [What to Do If an Installation Error Occurs?](#)
- [Reviewing the Log of an Installation Session](#)
- [Troubleshooting Configuration Assistants](#)
- [Silent-Mode Response File Error Handling](#)
- [Cleaning Up After a Failed Installation](#)

Verify Requirements

Before performing any of the troubleshooting steps in this appendix, ensure that the system meets the requirements and that you have completed all of the preinstallation tasks specified in [Chapter 2](#).

Read the Release Notes

Read the release notes for the product before installing it. The release notes are available on the Oracle Database 10g disc. The latest version of the release notes is also available on the Oracle Technology Network Web site:

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

What to Do If an Installation Error Occurs?

If you encounter an error during installation:

- Do not exit Oracle Universal Installer.
- If you clicked **Next** after you entered incorrect information on one of the installation screens, 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, refer to the "[Reviewing the Log of an Installation Session](#)" section on page F-2.
- If you encounter an error while a configuration assistant is running, refer to the "[Troubleshooting Configuration Assistants](#)" section on page F-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 F-4.

Reviewing the Log of an Installation Session

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

To view the log file, follow these steps:

1. If necessary, enter the following command to determine the location of the `oraInventory` directory:

```
$ cat /var/opt/oracle/oraInst.loc
```

The `inventory_loc` parameter in this file specifies the location of the `oraInventory` directory.

2. Enter the following command to change directory to Oracle Universal Installer log file directory, where `orainventory_location` is the location of the `oraInventory` directory:

```
$ cd /orainventory_location/logs
```

3. Enter the following command to determine the name of the log file:

```
$ ls -ltr
```

This command lists the files in the order of creation, with the most recent file shown last. Installer log files have names similar to the following, where `date_time` indicates the date and time that the installation started:

```
installActionsdate_time.log
```

4. To view the most recent entries in the log file, where information about a problem is most likely to appear, enter a command similar to the following:

```
$ tail -50 installActionsdate_time.log | more
```

This command displays the last 50 lines in the log file.

5. If the error displayed by Oracle Universal Installer or listed in the log file indicates a relinking problem, refer to the following file for more information:

```
$ORACLE_HOME/install/make.log
```

Troubleshooting Configuration Assistants

To troubleshoot an installation error that occurs when a configuration assistant is running:

- Review the installation log files listed in the ["Reviewing the Log of an Installation Session"](#) section on page F-2.
- Review the specific configuration assistant log file located in the `$ORACLE_HOME/cfgtoollogs` directory. Try to fix the issue that caused the error.
- If you see the "Fatal Error. Reinstall" message, then look for the cause of the problem by reviewing the log files. Refer to the ["Fatal Errors"](#) section on page F-3 for further instructions.

Configuration Assistant Failure

Oracle configuration assistant failures are noted at the bottom of the installation screen. The configuration assistant interface displays additional information, if available. The configuration assistant execution status is stored in the following file:

oraInventory_location/logs/installActionsdate_time.log

The execution status codes are listed in the following table:

Status	Result Code
Configuration assistant succeeded	0
Configuration assistant failed	1
Configuration assistant cancelled	-1

Fatal Errors

If you receive a fatal error while a configuration assistant is running, then you must remove the current installation and reinstall the Oracle software, as follows:

1. Remove the failed installation as described in the "[Cleaning Up After a Failed Installation](#)" section on page F-4.
2. Correct the cause of the fatal error.
3. Reinstall the Oracle software.

Silent-Mode Response File Error Handling

To determine whether a silent-mode installation succeeds or fails, refer to the following log file:

/oraInventory_location/logs/silentInstalldate_time.log

If necessary, refer to the previous section for information about determining the location of the *oraInventory* directory.

A silent installation fails if:

- You do not specify a response file
- You specify an incorrect or incomplete response file

For example, a common problem is that while all the product-specific data is filled out correctly, the staging area location may be incorrect. If this is the case, then check the `FROM_LOCATION` variable and ensure that it points to the `products.xml` file in the installation media. The `products.xml` file is in the `response/stage` directory in the installation media.
- Oracle Universal Installer encounters an error, such as insufficient disk space

Oracle Universal Installer or configuration assistant validates the response file at run time. If the validation fails, the silent-mode installation or configuration process ends. Oracle Universal Installer treats values for parameters that are of the wrong context, format, or type as if no value was specified in the file.

Cleaning Up After a Failed Installation

If an installation fails, then you must remove files that Oracle Universal Installer creates during the attempted installation and remove the Oracle home directory. Perform the following steps to remove the files:

1. Start Oracle Universal Installer as described in the "[Installing the Oracle Database Software](#)" section on page 3-4.
2. Click **Deinstall Products** on the Welcome window or click **Installed Products** on any Installer window.

The Inventory window appears, listing installed products.

3. Select the Oracle home that contains the products to be removed, then click **Remove**.
4. Manually remove the Oracle home directory created during the failed installation.
5. Reinstall the Oracle software.

Frequently Asked Questions About Installation

Use the following guidelines to decide how to install Oracle Database components:

- [Installing Oracle Database or Oracle Database Client](#)
- [Installing Oracle Database Tools](#)
- [Installing Oracle Database with Oracle Applications](#)

Installing Oracle Database or Oracle Database Client

- [I only need one instance of Oracle Database or I just want to install a test database to get familiar with the product. How do I install Oracle Database for these situations?](#)
- [How can I create an Oracle database that can handle transaction-heavy or data warehousing applications?](#)
- [What's the best way to install multiple Oracle databases?](#)
- [How do I configure client connections to an Oracle database?](#)
- [What is the best way to install Oracle Database Client if my client nodes have limited disk space?](#)
- [How do I upgrade Oracle Database?](#)
- [How do I migrate my non-Oracle databases to Oracle Database?](#)

I only need one instance of Oracle Database or I just want to install a test database to get familiar with the product. How do I install Oracle Database for these situations?

For installation procedure and special requirements, refer to *Oracle Database Installation Guide for Apple Mac OS X (Intel)*.

How can I create an Oracle database that can handle transaction-heavy or data warehousing applications?

If you want to create a starter database designed for transaction-heavy or data warehousing applications, then refer to *Oracle Database Installation Guide for Apple Mac OS X (Intel)*. Select the **Advanced Installation** method, and then select the database type you want on the Select Database Configuration screen.

See Also: *Oracle Database Data Warehousing Guide* after installation

Alternatively, you can install Oracle OLAP during the Oracle Database installation. Oracle OLAP provides optimal support for database environments that must meet OLAP requirements. To do so, select **Advanced Installation**, then **Custom**, and on the Available Product Components screen, select **Oracle OLAP**.

See Also:

- *Oracle OLAP Application Developer's Guide*
- *Oracle OLAP Reference*
- *Oracle OLAP DML Reference*
- *Oracle OLAP Java API Reference*
- *Oracle OLAP Analytic Workspace Java API Reference*

What's the best way to install multiple Oracle databases?

Use *Oracle Database Installation Guide for Apple Mac OS X (Intel)* to install Oracle Database using either of the following methods:

- **Installing with response files:** This method lets you run Oracle Universal Installer at a command line using a response file that contains settings specific to each computer.
- **Cloning an existing Oracle home:** Install Oracle Database in one computer using interactive mode. Afterwards, you can clone its existing Oracle home in each location and then create a new database from there. You can also clone databases, by referring to *Oracle Database Administrator's Guide*.

How do I configure client connections to an Oracle database?

1. Install Oracle Database on a server by using *Oracle Database Installation Guide for Apple Mac OS X (Intel)*.
2. Use *Oracle Database Client Installation Guide for Apple Mac OS X (Intel)* to install Oracle Database Client on each client node.

If you have many client nodes, then consider staging the software centrally, mapping the drive, and running Oracle Universal Installer in noninteractive mode.

What is the best way to install Oracle Database Client if my client nodes have limited disk space?

1. Install Oracle Database onto a server by using *Oracle Database Installation Guide for Apple Mac OS X (Intel)*.
2. Use *Oracle Database Client Installation Guide for Apple Mac OS X (Intel)* to install Oracle Database Client on each client node, and select the Instant Client installation type.

If you have many client nodes, then consider running Oracle Universal Installer in noninteractive mode.

How do I upgrade Oracle Database?

Refer to *Oracle Database Upgrade Guide*.

See Also: *Oracle Database Administrator's Guide* if you want to use software cloning to upgrade Oracle Database

How do I migrate my non-Oracle databases to Oracle Database?

Use Oracle Migration Workbench to migrate your non-Oracle databases and applications to Oracle. Oracle Migration Workbench software and documentation are available at:

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

Installing Oracle Database Tools

- [How do I install Oracle Application Server?](#)
- [How can I administer and monitor my Oracle Database products?](#)
- [How do I manage security for my Oracle Database products?](#)
- [Does Oracle Database provide OLAP tools so that I can analyze data such as trends and time series in my database?](#)
- [Does Oracle Database provide data mining tools that I can use to discover hidden meaning in my data and predict likely outcomes based on my data?](#)
- [How do I perform backup and recovery operations for Oracle Database?](#)

How do I install Oracle Application Server?

Refer to *Oracle Application Server Installation Guide*. How you install Application Server depends on whether you already have Oracle Database installed:

- If you do not have Oracle Database installed or you do not want Oracle Application Server to use any of your existing Oracle Databases, then Oracle Universal Installer lets you install Oracle Application Server with its own Oracle Database. This database is populated with the metadata that Oracle Application Server needs for running.
- If you want Oracle Application Server to use an existing Oracle Database, then do the following:
 1. From the Oracle Application Server installation media, run Oracle Application Server Repository Creation Assistant to populate your database with the metadata that Application Server needs.
 2. Install the remaining Oracle Application Server components by following the instructions in the *Oracle Application Server Installation Guide*.

How can I administer and monitor my Oracle Database products?

To perform regular administrative functions such as creating, configuring, or deleting databases, or managing database templates, use one of the following methods:

To manage only the single database and listener that you are installing:

1. Use *Oracle Database Installation Guide for Apple Mac OS X (Intel)* to install Oracle Database.
2. From Oracle Database, use Database Configuration Assistant to manage your databases.

How do I manage security for my Oracle Database products?

Oracle provides a wide range of security solutions for your enterprise environment, including centralized administration and security features integrated with Oracle Internet Directory. The set of Oracle security services called Oracle Platform Security integrates the security features built into Oracle Database, Oracle Application Server,

and the Oracle Identity Management infrastructure. Combined, these features enable the development and deployment of secure e-business applications.

Oracle Identity Management includes Oracle Internet Directory, a centralized repository that simplifies administration of users and applications in the Oracle environment by means of the following components:

- Oracle Internet Directory client tools, including LDAP command-line tools, the Oracle Internet Directory SDK, and Oracle Directory Manager.
- Oracle Internet Directory server components, including the directory server, the directory replication server, the directory integration server, and various tools for starting and stopping them.

Oracle Database includes the Oracle Internet Directory client tools, but not the Oracle Internet Directory server components.

See Also:

- *Oracle Application Server Installation Guide* (to install Oracle Identity Management)
- *Oracle Database Security Guide*
- *Oracle Database Advanced Security Administrator's Guide*
- *Oracle Database Enterprise User Administrator's Guide*
- *Oracle Label Security Administrator's Guide*
- *Oracle Application Server Security Guide*
- <http://www.oracle.com/technology/deployp/security/index.html> for Oracle Technology Network topics on database security

Does Oracle Database provide OLAP tools so that I can analyze data such as trends and time series in my database?

Yes, install Oracle OLAP, which is provided in the Oracle Database installation. Oracle OLAP provides optimal support for database environments that must meet OLAP requirements.

Use either of the following methods in *Oracle Database Installation Guide for Apple Mac OS X (Intel)* to install Oracle OLAP:

- Select the **Custom** installation type, and in the Available Product Components screen, select **Oracle OLAP**.

See Also:

- *Oracle OLAP Application Developer's Guide*
 - *Oracle OLAP Reference*
 - *Oracle OLAP DML Reference*
 - *Oracle OLAP Java API Reference*
 - *Oracle OLAP Analytic Workspace Java API Reference*
- Select the **Enterprise Edition** installation type, and then on the Select Database Configuration screen, select the **Data Warehouse** configuration.

See Also: *Oracle Database Data Warehousing Guide* after installation

Does Oracle Database provide data mining tools that I can use to discover hidden meaning in my data and predict likely outcomes based on my data?

Yes. Install Oracle Data Mining, which is provided in the Oracle Database installation. With the Oracle Data Mining option, you can create and execute predictive and descriptive data mining models that use a variety of algorithms.

Use the following method in *Oracle Database Installation Guide for Apple Mac OS X (Intel)* to install Oracle Data Mining:

1. Select the **Enterprise Edition** installation type.
2. On the Select Database Configuration screen, select the **General Purpose** configuration.

If you want the Database to execute predefined mining models, but not support model creation, then install the Data Mining Scoring Engine instead of Oracle Data Mining. Use the following method in *Oracle Database Installation Guide for Apple Mac OS X (Intel)* to install the Data Mining Scoring Engine:

1. Select the **Custom** installation type.
2. In the Available Product Components screen, select **Data Mining Scoring Engine**.

See Also: The following manuals after you have installed Oracle Data Mining:

- *Oracle Data Mining Concepts*
- *Oracle Data Mining Administrator's Guide*
- *Oracle Data Mining Application Developer's Guide*
- *Oracle Data Mining Java API Reference*
- *Oracle Database PL/SQL Packages and Types Reference* (search for "Data Mining")
- *Oracle Database SQL Reference* (search for Data Mining)

How do I perform backup and recovery operations for Oracle Database?

Use Oracle Database Recovery Manager (RMAN), which is a backup and recovery tool integrated into Oracle Database. This tool satisfies the pressing demands of high-performance, manageable backup, and recovery. Recovery Manager is native to the database server, automatically tracks database structure changes, and optimizes operations accordingly. In addition, Recovery Manager is integrated with leading tape media management products, so that Oracle database backups can be integrated with your existing networked data protection infrastructure.

See Also:

- *Oracle Database Backup and Recovery Basics*
- *Oracle Database Backup and Recovery Advanced User's Guide*
- *Oracle Database Backup and Recovery Reference*
- *Oracle Database Backup and Recovery Quick Start Guide*

Installing Oracle Database with Oracle Applications

- [How do I install my Oracle applications with Oracle Database?](#)
- [How can I migrate my non-Oracle applications to Oracle?](#)

How do I install my Oracle applications with Oracle Database?

In most cases, install Oracle Database itself, then install the Oracle application. The Oracle Universal Installer for that application prompts you for the connection information. Check the application documentation requirements.

How can I migrate my non-Oracle applications to Oracle?

Use Oracle Migration Workbench to migrate your non-Oracle applications to Oracle. Oracle Migration Workbench software and documentation are available at:

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

Index

A

accounts
 reviewing, 5-1
 unauthenticated access to, 5-4
aliases, multiple on computers, 2-5
ANONYMOUS user
 unauthenticated account access with, 5-4
applications, migrating non-Oracle applications to
 Oracle, G-6

B

backups of database
 Oracle Database Recovery Manager, G-5
base directory
 See Oracle base directory
Bash shell
 default user startup file, 2-21
.bash_profile file, 2-21
Basic installation type
 noninteractive installations, A-5
Bourne shell
 default user startup file, 2-21

C

C compiler
 requirement on Mac OS X, 2-3
C shell
 default user startup file, 2-21
certification, hardware and software, 1-5
chmod command, 2-17, 2-19
chown command, 2-17, 2-19
CLASSPATH environment variable, 4-7
client static library, generating, 4-3
cloning
 Oracle home, B-1
computers with multiple aliases, 2-5
computers, non-networked, 2-5
configuration assistants
 failure, F-3
 troubleshooting, F-2
configuring
 kernel parameters, 2-11
Connection Manager

 ports, ranges and protocol, D-2
control files
 locating, 5-6
 naming, C-6
 reviewing, 5-5
CONTROL_FILES initialization parameter, 5-6
Custom installation type
 reasons for choosing, 2-6
custom.rsp file, A-4

D

data files
 creating separate directories for, 2-19
 defined, 5-5
 minimum disk space for, 2-18
 naming, C-6
 options for placing on file system, 2-17
 recommendations for file system, 2-18
 reviewing, 5-5
 setting permissions on data file directories, 2-19
 setting up, 5-5
Data Guard
 ports, ranges and protocol, D-2
Data Mining Scoring Engine
 about, G-5
 installing, G-5
Data Mining Scoring Engine, installation
 guidelines, 3-2
data mining tools
 Data Mining Scoring Engine, G-5
 Oracle Data Mining, G-5
data warehousing tool
 Oracle OLAP, G-4
Database Configuration Assistant
 running in silent mode, A-8
 troubleshooting, F-2
databases
 administrative information, C-2
 files, 5-5
 identifying, 5-4
 initialization parameter file, 5-5
 OLAP support (Oracle OLAP), G-4
 Optimal Flexible Architecture file naming
 conventions, C-6
 recovery with Oracle Backup and Recovery, G-5

- redo log files, 5-6
- removing, 6-1
- security management, G-3
- tablespaces, 5-5
- DB_DOMAIN initialization parameter, 5-4
- DB_NAME initialization parameter, 5-4
- dba group
 - and SYSDBA privilege, 2-6
 - creating, 2-8, 2-9
 - description, 2-6
- dbca.rsp file, A-4
- DBSNMP user password, 3-4
- default file mode creation mask
 - setting, 2-21
- DHCP computers, installing on, 2-4
- directory
 - creating separate data file directories, 2-19
 - database file directory, 2-18
 - Oracle base directory, 2-13
 - Oracle home directory, 2-14
 - Oracle Inventory directory, 2-14
 - oraInventory, 2-14
 - permission for data file directories, 2-19
- disk mirroring, C-9
- disk space
 - checking, 2-2
 - requirement for Oracle base directory, 2-16
- disk striping, C-9
- DISPLAY environment variable
 - setting, 2-21
- DOMAIN_NAME initialization parameter, 5-4
- drive failure safeguards, C-2
- Dynamic Host Configuration Protocol. *See* DHCP

E

- enterprise.rsp file, A-4
- env command, 2-23
- environment
 - checking settings, 2-23
 - configuring for oracle user, 2-21
- environment variables
 - DISPLAY, 2-21
 - NLS_LANG, E-1
 - ORACLE_BASE, 2-14, 2-17, 2-21
 - ORACLE_HOME, 2-20, 2-21, 2-22
 - ORACLE_HOSTNAME, 2-5
 - ORACLE_SID, 2-21
 - PATH, 2-21
 - removing from shell startup file, 2-21
 - SHELL, 2-21
 - TMP and TMPDIR, 2-22
 - TNS_ADMIN, 2-22
- errors
 - configuration assistants, F-2
 - installation, F-1, F-2, F-3
 - noninteractive installation, F-3
 - silent mode, F-3
- /etc/sysctl.conf file, 2-12
- examples

- Oracle base directories, 2-14
- external jobs
 - operating system user required for, 2-6
- extjob executable
 - operating system user required for, 2-6

F

- FAQ for installation, G-1 to ??
- fatal errors, F-3
- file mode creation mask
 - setting, 2-21
- file naming, C-2
- file sets, 2-3
- file system
 - appropriate for Oracle base directory, 2-17
 - data file and recovery file placement
 - options, 2-17
 - organization, C-1
 - requirements for Oracle base directory, 2-17
 - using for data files, 2-18
- files, C-6
 - \$ORACLE_HOME/dbs/initsid.ora, 5-5
 - .bash_profile, 2-21
 - control, C-6
 - control, 5-6
 - custom.rsp, A-4
 - data files, C-6
 - dbca.rsp, A-4
 - editing shell startup file, 2-21
 - enterprise.rsp, A-4
 - /etc/group, C-5
 - /etc/passwd, C-5
 - /etc/sysctl.conf, 2-12
 - listener.ora, 4-4
 - .login, 2-21
 - mgw.ora, 4-7
 - oraInst.loc, 2-7, 2-15
 - oraInst.loc file, A-3
 - oratab, 2-15
 - .profile, 2-21
 - redo log, 5-6
 - response files, A-3
 - standard.rsp, A-4
 - tnsnames.ora, 4-4
 - /var/opt/oracle/oraInst.loc, F-2
- frequently asked installation questions, G-1 to ??

G

- global database name, 5-4
- globalization support, E-1
- group file, C-5
- groups
 - checking for existing oinstall group, 2-7
 - creating the dba group, 2-8, 2-9
 - creating the oinstall group, 2-7

H

- hardware certification, 1-5

- hardware requirements, 2-2
- home directory
 - distribution, C-2
 - login, C-2
 - See Oracle home directory
- host name, setting before installation, 2-5

I

- initialization parameter file
 - description, 5-5
 - in databases, 5-5
 - initsid.ora, 5-5
- initialization parameters
 - DB_NAME, 5-4
 - DOMAIN_NAME, 5-4
 - SERVICE_NAMES, 5-4
- initsid.ora file, 5-5
- initsid.ora initialization parameter file, 5-5
- installation
 - accessing installation software, 3-2
 - available products, 1-3
 - cleaning up after a failed installation, F-4
 - completing, 3-6
 - component-specific guidelines, 3-1
 - computer aliases, multiple, 2-5
 - considerations, 1-5
 - errors, F-2, F-3
 - silent mode, F-3
 - FAQ for Oracle Database products, G-1 to ??
 - guidelines, 3-4
 - laptops, 2-5
 - log files, F-1
 - noninteractive
 - error handling, F-3
 - oraInst.loc file, A-3
 - quick installation, G-1
 - reinstalling Oracle software, 3-2
 - response files, A-1, A-3
 - preparing, A-3, A-5
 - silent mode, F-3
 - templates, A-3
 - silent mode, A-6, A-7
 - upgrading, G-2
 - with other components, G-1 to ??
- installation errors
 - steps to resolve, F-1
- installation guidelines, 3-4, 3-5
 - Oracle Universal Installer, 3-4
- installation overview, 1-1
- installation software
 - extracting, 3-3
- installation software, accessing, 3-2
- instance
 - instance identifier (SID), 2-21
- I/O loads, C-3
 - Optimal Flexible Architecture, C-2
- IP addresses, multiple, 2-4
- IPC protocol address
 - Oracle Messaging Gateway setting, 4-6

- iSQL*Plus
 - ports, changing, D-3
 - ports, ranges and protocol, D-2

J

- Java Libraries for Oracle JVM and Oracle interMedia
 - postinstallation tasks, 4-5
- JDK
 - internationalization class, 4-7
 - run-time class, 4-7
- JDK requirements, 2-3

K

- kern.corefile parameter
 - recommended value, 2-11
- kernel parameters
 - changing, 2-12
 - configuring, 2-11
- kern.maxfiles parameter
 - recommended value, 2-11
- kern.maxfilesperproc parameter
 - recommended value, 2-11
- kern.sysv.maxproc parameter
 - recommended value, 2-11
- kern.sysv.maxprocpid parameter
 - recommended value, 2-11
- kern.sysv.semmani parameter
 - recommended value, 2-11
- kern.sysv.semman parameter
 - recommended value, 2-11
- kern.sysv.semman parameter
 - recommended value, 2-11
- kern.sysv.semman parameter
 - recommended value, 2-11
- kern.sysv.shmall parameter
 - recommended value, 2-11
- kern.sysv.shmmmax parameter
 - recommended value, 2-11
- kern.sysv.shmmni parameter
 - recommended value, 2-11
- Korn shell
 - default user startup file, 2-21
- ksh
 - See Korn shell

L

- languages
 - installing Oracle components in different languages, E-2
- laptops, installing Oracle Database on, 2-5
- licensing information, 1-3
- listener
 - identifying Oracle home for, 2-20
 - lsnrctl command, 2-21
 - stopping, 2-20, 2-21
 - stopping existing listener process, 2-20

- listener.ora file, 4-4
 - modifying for external procedures, 4-6
- local device
 - using for data files, 2-18
- log files, F-1
 - troubleshooting, F-2
- logical volume manager
 - See LVM
- .login file, 2-21
- loopback adapters
 - non-networked computers, 2-5
- lsnrctl command, 2-21

M

- Mac OS X
 - setting shell limits, 2-13
- mask
 - setting default file mode creation mask, 2-21
- memory requirements, 2-2
- MGW_AGENT service name, 4-7
- mgwextproc service
 - adding static service information, 4-6
- mgw.ora file
 - modifying, 4-7
- migrating applications to Oracle, G-6
- migrating non-Oracle databases to Oracle, G-3
- mkdir command, 2-17, 2-19
- mode
 - setting default file mode creation mask, 2-21
- mount point
 - for Oracle base directory, 2-14
- mount points
 - Optimal Flexible Architecture conventions for creating, C-3
- MQSeries
 - class, 4-7
- multihomed computers, installing on, 2-4
- multiple aliases, computers with, 2-5
- multiple Oracle homes, 1-6

N

- naming subdirectories, C-5
- Net Configuration Assistant
 - troubleshooting, F-2
- Net Configuration Assistant (NetCA)
 - response files, A-7
 - running at command prompt, A-7
- netca.rsp file, A-4
- net.inet.ip.portrange.first parameter
 - recommended value, 2-11
- net.inet.ip.portrange.last parameter
 - recommended value, 2-11
- network adapters
 - computers with multiple aliases, 2-5
 - non-networked computers, 2-5
 - primary, on computers with multiple aliases, 2-5
 - See also loopback adapters, primary network adapters

- network cards, multiple, 2-4
- network setup
 - about, 2-3
 - computers with multiple aliases, 2-5
- network topics
 - DHCP computers, 2-4
 - laptops, 2-5
 - multiple network cards, 2-4
 - non-networked computers, 2-5
- NLS_LANG environment variable, E-1
- NLS_LANG parameter
 - about, E-1
- nobody user
 - description, 2-6
- noninteractive installation
 - oraInst.loc file, A-3
 - response files
 - preparing, A-3, A-5
 - templates, A-3
 - silent mode, A-6, A-7
 - errors, F-3
- noninteractive mode
 - about, A-1
 - reasons for using, A-2
 - See also response files, silent mode, A-1
- non-networked computers, 2-5

O

- OCCI
 - postinstallation tasks, 4-8
- oinstall group
 - checking for existing, 2-7
 - creating, 2-7
 - description, 2-6
- OLAP tools
 - about, G-4
 - Oracle OLAP, G-4
- OMF
 - See Oracle Managed Files
- oper group
 - and SYSOPER privilege, 2-6
 - creating, 2-9
 - description, 2-6
- operating system groups
 - creating the dba group, 2-9
 - creating the oinstall group, 2-7
 - oinstall, 2-6
 - OSDBA, 2-6
 - OSOPER, 2-6
 - osoper, 2-6
 - requirements, 2-6
- operating system requirements, 2-3
- operating system users
 - creating the oracle user, 2-9
 - nobody, 2-6
 - oracle, 2-6
 - requirements, 2-6
 - root user, 3-5
 - unprivileged user, 2-6

- Optimal Flexible Architecture
 - administrative information, C-2
 - changes for Oracle Database 10g, C-3
 - characteristics of an installation compliant with, C-1
 - concurrent execution of application software, C-2
 - conventions for creating mount points, C-3
 - distributed I/O loads, C-2
 - drive failure safeguards, C-2
 - file identification, C-7
 - file mapping, C-2, C-8
 - file naming, C-2
 - file system organization, C-1
 - files systems, C-4
 - hardware support, C-2
 - home directory distribution, C-2
 - I/O loads, C-3
 - login home directories, C-2
 - naming, C-3
 - database files, C-6
 - Oracle base directory, C-4
 - subdirectories, C-5
 - tablespaces, C-7
 - very large databases, C-4
 - Oracle Managed Files, C-6
 - pathnames, C-5
 - recommendations for Oracle base directory, 2-13
 - recommended path for Oracle base directory, 2-13
 - recommended path for Oracle home directory, 2-14
 - recommended path for Oracle Inventory directory, 2-14
 - special tablespaces, C-7
 - standard, C-1
 - tablespace contents, C-2
 - UNIX directory subtrees, C-2
 - using separate segments, C-6
- Oracle Application Server, G-3
- Oracle applications
 - installing with Oracle Database, G-6
- Oracle base directory
 - and ORACLE_BASE environment variable, 2-14
 - creating, 2-17
 - creating new, 2-16
 - description, 2-13
 - determining disk space on, 2-16
 - disk space requirements, 2-16
 - equivalent directory on Microsoft Windows, 2-14
 - examples, 2-14
 - identifying appropriate file system, 2-17
 - identifying existing, 2-15
 - mount point for, 2-14
 - naming conventions, C-4
 - recommended path, 2-13
 - relationship with Oracle software owner user, 2-14
 - requirement for, 2-13
 - requirements for existing directory, 2-16
 - requirements on file system, 2-17
- Oracle Connection Manager, installation
 - guidelines, 3-2
- Oracle Data Mining
 - about, G-5
 - installing, G-5
- Oracle Database
 - administering and monitoring, G-3
 - creating data file directories, 2-19
 - Custom installation, 1-3
 - Enterprise Edition installation, 1-3
 - installing with Oracle applications, G-6
 - installing with other Oracle components, G-1 to ??
 - minimum disk space requirements, 2-18
 - privileged groups, 2-6
 - quick installation, G-1
 - security management, G-3
 - setting ORACLE_SID environment variable, 2-21
 - Standard Edition installation, 1-3
 - upgrading, G-2
- Oracle Database Client
 - configuring connections, G-2
- Oracle Database components
 - administering and monitoring, G-3
 - FAQ on installing, G-1 to G-3
 - installing with Oracle applications, G-6
 - installing with Oracle Database tools, G-3
- Oracle Database Configuration Assistant
 - response file, A-4
- Oracle Database Recovery Manager (RMAN)
 - about, G-5
- Oracle Event Manager
 - ports, ranges and protocol, D-3
- Oracle home
 - cloning, B-1
- Oracle home directory
 - description, 2-14
 - identifying for listener, 2-20
 - multiple homes, network considerations, 2-4
 - recommended path, 2-14
 - requirement for, 2-14
 - requirements, 2-14
 - using to identify Oracle base directory, 2-15
- Oracle home name, 2-14
- Oracle host name, setting before installation, 2-5
- Oracle internationalization class, 4-7
- Oracle Internet Directory, G-4
- Oracle Inventory
 - description, 2-14
 - pointer file, 2-7
- Oracle Inventory directory
 - description, 2-14
 - recommended path, 2-14
- Oracle Inventory group
 - checking for existing, 2-7
 - creating, 2-7, 2-8
 - description, 2-6
- Oracle JDBC class, 4-7
- Oracle Label Security
 - installation guidelines, 3-2

- postinstallation tasks, 4-5
- Oracle Managed Files
 - Optimal Flexible Architecture naming conventions, C-6
- Oracle Messaging Gateway
 - postinstallation tasks, 4-5
- Oracle Messaging Gateway class, 4-7
- Oracle Migration Workbench
 - migrating non-Oracle applications to Oracle, G-6
 - migrating non-Oracle databases to Oracle, G-3
- Oracle Net
 - configuration file directory, 4-4
 - identifying Oracle home for listener, 2-20
 - lsnrctl command, 2-21
 - stopping existing listener, 2-20
 - stopping listener, 2-20
 - stopping the listener, 2-21
- Oracle Net Configuration Assistant
 - response file, A-4
- Oracle Net Services
 - postinstallation tasks, 4-4
- Oracle OLAP
 - about, G-4
- Oracle Precompilers
 - postinstallation tasks, 4-8
- Oracle software
 - removing, 6-2
- Oracle Software Owner user
 - creating, 2-9
- Oracle software owner user
 - configuring environment for, 2-21
 - creating, 2-9
 - description, 2-6
 - determining default shell, 2-21
 - relationship with Oracle base directory, 2-14
 - setting shell limits for, 2-13
- Oracle SQL*Net Listener
 - ports, ranges and protocol, D-2
- Oracle Text supplied knowledge bases
 - postinstallation tasks, 4-5
- Oracle Ultra Search
 - ports, changing, D-4
 - ports, ranges and protocol, D-3
- Oracle Universal Installer
 - guidelines for using, 3-1
 - installation guidelines, 3-1, 3-4
 - response files, A-1
 - list of, A-4
 - running, 3-4
 - running in different languages, E-2
- oracle user
 - configuring environment for, 2-21
 - creating, 2-9, 2-10
 - description, 2-6
 - determining default shell, 2-21
 - relationship with Oracle base directory, 2-14
 - setting shell limits for, 2-13
- Oracle XML DB
 - ports, D-4
 - ports, changing, D-4
 - ports, ranges and protocol, D-3
- ORACLE_BASE environment variable, 2-14, 2-17
 - removing from shell startup file, 2-21
 - setting, 2-21
- ORACLE_HOME environment variable
 - removing from shell startup file, 2-21
 - setting, 2-20
 - unseting, 2-22
- ORACLE_HOSTNAME environment variable
 - about, 2-5
 - computers with multiple aliases, 2-5
 - multihomed computers, 2-5
 - setting before installation, 2-5
- ORACLE_SID environment variable
 - removing from shell startup file, 2-21
 - setting, 2-21
- oraInst.loc file, 2-15
 - location, 2-7
 - location of, 2-7
- oraInventory directory
 - See* Oracle Inventory directory
- oratab file, 2-15
 - formats, 2-15
 - location of, 2-15
- OSDBA group
 - and SYSDBA privilege, 2-6
 - creating, 2-8, 2-9
 - description, 2-6
- OSOPER group
 - and SYSOPER privilege, 2-6
 - description, 2-6
- OTN Web site
 - downloading installation software from, 3-2

P

- passwd file, C-5
- passwords
 - DBSNMP, 3-4
 - guidelines, 3-4
 - resetting, 5-3
 - with SQL*Plus, 5-4
 - reviewing, 5-1
 - specifying for response files, A-2
 - SYS, 3-4
 - SYSMAN, 3-4
 - SYSTEM, 3-4
 - unlocking, 5-3
 - with SQL*Plus, 5-4
 - See also* security
- PATH environment variable
 - setting, 2-21
- pathnames
 - Optimal Flexible Architecture, C-5
- permissions
 - for data file directories, 2-19
 - for Oracle base directory, 2-17
- port numbers
 - managing, D-1
- portlist.ini file, D-2

- ports
 - access URLs, D-2
 - configured for applications, D-2
 - Connection Manager, ranges and protocol, D-2
 - Data Guard, ranges and protocol, D-2
 - default ranges, D-1
 - iSQL*Plus, changing, D-3
 - iSQL*Plus, ranges and protocol, D-2
 - Oracle Event Manager, ranges and protocol, D-3
 - Oracle SQL*Net Listener, ranges and protocol, D-2
 - Oracle Ultra Search, changing, D-4
 - Oracle Ultra Search, ranges and protocol, D-3
 - Oracle XML DB, D-4
 - Oracle XML DB, changing, D-4
 - Oracle XML DB, ranges and protocol, D-3
- post-installation
 - required tasks, 4-1
 - patches, installing and downloading, 4-1
- postinstallation
 - recommended tasks
 - client static library, generating, 4-3
 - root.sh script, backing up, 4-2
 - user accounts, setting up, 4-3
 - required tasks
 - configuring OCCI, 4-8
 - configuring Oracle Messaging Gateway, 4-5
 - installing natively compiled Java Libraries for Oracle JVM and Oracle interMedia, 4-5
 - installing Oracle Text supplied knowledge bases, 4-5
 - Oracle Label Security, configuring, 4-5
 - Oracle Net Services, configuring, 4-4
 - Oracle Precompilers, 4-8
- Precompilers
 - requirements on Mac OS X, 2-3
- privilege, 2-6
- privileged groups
 - for Oracle Database, 2-6
- Pro*C/C++
 - configuring, 4-8
 - requirements on Mac OS X, 2-3
 - See also* C compiler
- process
 - stopping existing, 2-20
 - stopping existing listener process, 2-20
 - stopping listener process, 2-20
- .profile file, 2-21
- ps command, 2-20

R

- RAID
 - using for Oracle data files, 2-18
- RAM requirements, 2-2
- readme.txt file, D-2
- recovery files
 - options for placing on file system, 2-17
- recovery of databases
 - Oracle Backup and Recovery, G-5

- Red Hat
 - operating system requirements, 2-3
- redo log, C-6
- redo log files
 - in starter database, 5-6
 - locating, 5-6
 - naming, C-6
 - reviewing, 5-5
- redundant array of independent disks
 - See* RAID
- reinstalling Oracle software, 3-2
- requirements
 - hardware, 2-2
- response files
 - about, A-1
 - creating with template, A-3
 - custom.rsp, A-4
 - dbca.rsp, A-4
 - enterprise.rsp, A-4
 - general procedure, A-2
 - Net Configuration Assistant, A-7
 - netca.rsp, A-4
 - passing values at command line, A-2
 - passwords, A-2
 - security, A-2
 - specifying with Oracle Universal Installer, A-6
 - standard.rsp, A-4
 - See also* silent mode, noninteractive mode, A-1
- response files installation
 - about, A-1
- roadmap for installing Oracle Database
 - components, G-1 to ??
- root user, 3-5
 - logging in as, 2-1
- root.sh script
 - backing up, 4-2

S

- security
 - management tools, G-3
 - See also* passwords
- SERVICE_NAMES initialization parameter, 5-4
- shell
 - determining default shell for oracle user, 2-21
- SHELL environment variable
 - checking value of, 2-21
- shell limits
 - setting, 2-13
 - setting on Mac OS X, 2-13
- shell startup file
 - editing, 2-21
 - removing environment variables, 2-21
- SID, 5-5
 - setting ORACLE_SID environment variable, 2-21
- silent mode
 - about, A-1
 - reasons for using, A-2
 - See also* noninteractive mode, response files, A-1
- silent mode installation, A-6, A-7

- software certification, 1-5
- software requirements, 2-3
 - checking software requirements, 2-3
- SQLJ class, 4-7
- standard.rsp files, A-4
- startup file
 - for shell, 2-21
- static service information
 - adding for mgwextproc service, 4-6
- suppressed mode
 - reasons for using, A-2
- suppressed mode. *See* noninteractive mode
- SUSE
 - operating system requirements, 2-3
- swap space
 - requirements, 2-2
- SYS user password, 3-4
- sysctl command, 2-12
- sysctl.conf file, 2-12
- SYSDBA privilege
 - associated operating system group, 2-6
- SYSMAN user password, 3-4
- SYSOPER privilege
 - associated operating system group, 2-6
- System Identifier, 5-5
 - See* SID
- SYSTEM user password, 3-4

T

- tablespaces
 - defined, 5-5
 - in databases, 5-5
 - reviewing, 5-5
 - setting up, 5-5
 - Optimal Flexible Architecture
 - naming, C-7
 - special tablespaces, C-7
- temporary disk space
 - requirements, 2-2
- TMP environment variable
 - setting, 2-22
- TMPDIR environment variable
 - setting, 2-22
- TNS_ADMIN environment variable
 - unsetting, 2-22
- tnsnames.ora file, 4-4
 - adding a connect descriptor, 4-7
 - MGW_AGENT service name, 4-7
 - modifying for external procedures, 4-7
- troubleshooting, F-1
 - fatal errors, F-3

U

- umask, 2-23
- umask command, 2-21, 2-23
- UNIX commands, 2-23
 - chmod, 2-17, 2-19
 - chown, 2-17, 2-19

- env, 2-23
- mkdir, 2-17, 2-19
- ps, 2-20
- sysctl, 2-12
- umask, 2-21
- unset, 2-22
- unsetenv, 2-22
- UNIX directory subtrees, C-2
- UNIX groups
 - checking for existing oinstall group, 2-7
 - creating the dba group, 2-8
- UNIX users
 - creating the oracle user, 2-9
- UNIX workstation
 - installing from, 2-1
- unprivileged user
 - nobody user, 2-6
- unset command, 2-22
- unsetenv command, 2-22
- upgraded databases
 - configuring, 4-2
- users
 - creating the oracle user, 2-9
 - operating system nobody user, 2-6
 - Oracle software owner user, 2-6
- UTLRP.SQL
 - recompiling invalid SQL modules, 4-2

V

- /var/opt/oracle/oraInst.loc, F-2
- very large databases
 - Optimal Flexible Architecture naming mount points, C-4

W

- WebSphere MQ class, 4-7
- Windows
 - analogy for Oracle base directory, 2-14