Oracle® Database

Installation Guide

11*g* Release 1 (11.1) for Linux

B32002-05

September 2008

ORACLE®

Oracle Database Installation Guide, 11*g* Release 1 (11.1) for Linux

B32002-05

Primary Author: Namrata Bhakthavatsalam

Contributing Authors: Kevin Flood, Clara Jaeckel, Emily Murphy, Terri Winters, Reema Khosla

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

Contents

2 Oracle Database Preinstallation Requirements

3 Installing Oracle Database

4 Oracle Database Postinstallation Tasks

5 Getting Started with Oracle Database

6 Removing Oracle Software

A Installing and Configuring Oracle Database Using Response Files

B Cloning an Oracle Home

C Using NAS Devices

D Optimal Flexible Architecture

E Managing Oracle Database Port Numbers

F Configuring Oracle Database Globalization Support

G Troubleshooting

H Frequently Asked Questions About Installation

I Country Codes

Glossary

Index

Preface

This guide provides instructions about installing and configuring Oracle Database for Linux. This guide covers Optimal Flexible Architecture, Database Storage Options, and Database Configuration Options. This guide also talks about installing and configuring database using response files, globalization support, ports, and troubleshooting.

The preface contains the following topics:

Audience

Oracle Database Installation Guide for Linux guide is intended for anyone responsible for installing Oracle Database 11*g* Release 1 (11.1) on a single Linux x86 system. Additional installation guides for Oracle Database, Oracle Real Application Clusters, Oracle Clusterware, Oracle Database Examples, and Oracle Enterprise Manager Grid Control are available on the relevant installation media.

> **See Also:** *Oracle Database Installation Guide for Linux* to install Oracle Database using the default settings

Documentation Accessibility

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

`http://www.oracle.com/accessibility/`

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

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

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:
	`dd if=/dev/rdsk/c0t1d0s6 of=/dev/rst0 bs=10b \`
	`count=10000`

braces { }	Braces indicate required items: `.DEFINE {macro1}`
brackets [ ]	Brackets indicate optional items: `cvtcrt` *`termname`* `[`*`outfile`*`]`
ellipses ...	Ellipses indicate an arbitrary number of similar items: `CHKVAL fieldname value1 value2 ... valueN`
italics	Italic type indicates a variable. Substitute a value for the variable: *`library_name`*
vertical line |	A vertical line indicates a choice within braces or brackets: `FILE` *`filesize`* `[K|M]`

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.

This guide contains information required to install Oracle Database 11*g* Release 1 (11.1) on various platforms of Linux. Ensure that you review information related to the platform on which you intend to install Oracle Database 11*g*.

The platform-specific documentation for this product is available in both Adobe portable document format (PDF) and HTML format on the product media. To access the platform-specific documentation on media:

1. Use a Web browser to open the `welcome.htm` file in the top-level directory of the media.
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 11*g* products is available in both HTML and PDF formats in the Oracle Database 11*g* Release 1 (11.1) Online Documentation Library. To check for updates to this document and to view other Oracle documentation, select the Documentation link or the Software & Patches link on the Oracle Database 11*g* Release 1 Beta Program Web site.

Related Documentation

The platform-specific documentation for Oracle Database 11*g* products includes the following manuals:

- *Oracle Database Release Notes for Linux*
- *Oracle Database Client Installation Guide for Linux*
- *Oracle Database Examples Installation Guide*
- *Oracle Real Application Clusters Installation Guide for Linux and UNIX*
- *Oracle Database Quick Installation Guide for Linux x86-64*
- *Oracle Database Client Quick Installation Guide for Linux x86-64*
- *Oracle Enterprise Manager Grid Control Installation and Basic Configuration*
- *Oracle Database Administrator's Reference for Linux and UNIX*
- *Oracle Database Storage Administrator's Guide*
- *Oracle Clusterware Installation Guide for Linux*
- *Oracle Database Upgrade Guide*
- *Oracle Database 2 Day DBA*

For information about Oracle error messages, see *Oracle Database Error Messages*. Oracle error message documentation is available only in HTML. If you only have access to the Oracle Database 10*g* Release 2 (10.2) Online Documentation Library, then you can browse the error messages by range. Once you find the specific range, use your browser's "find in page" feature to locate the specific message. When connected to the Internet, you can search for a specific error message using the error message search feature of the Oracle online documentation.

Many books in the documentation set use the sample schemas of the seed database, which is installed by default when you install Oracle. Refer to *Oracle Database Sample Schemas* for information on how these schemas were created and how you can use them yourself.

Printed documentation is available for sale in the Oracle Store at:

http://oraclestore.oracle.com/

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network. You must register online before using OTN; registration is free and can be done at:

http://www.oracle.com/technology/membership/

If you already have a user name and password for Oracle Technology Network, then you can go directly to the documentation section of the Oracle Technology Network Web site at:

http://www.oracle.com/technology/documentation/

Refer to *Oracle Database Release Notes for Linux* or important information that was not available when this book was released. The release notes for Oracle Database 11*g* are updated regularly. The most recent version is available on 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.
italic	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.

What's New in Oracle Database 11*g*

The following is a list of new features or enhancements provided with Oracle Database 11*g*:

New Components Available for Installation

The following are the new components available while installing Oracle Database 11*g*:

- **Oracle Application Express**: This feature is installed with Oracle Database 11*g*. It was previously named HTML DB, and was available as a separate Companion CD component. Oracle Application Express is now installed by default with any Oracle Database 11*g* installation.
- **Oracle Configuration Manager**: This feature is offered during installation. It was previously named Customer Configuration repository (CCR). It is an optional component for database installation and can be installed with any Oracle Database 11*g* installation. Oracle Configuration Manager gathers and stores details relating to the configuration of the software stored in database Oracle home directories.
- **Oracle Database Vault**: This feature is installed with Oracle Database 11*g*. It is an optional component for database installation.
- **Oracle Real Application Testing**: This feature is installed by default with the Enterprise Edition installation type of Oracle Database 11*g*.
- **Oracle SQL Developer**: This feature is installed by default with template-based database installations, such as General Purpose/Transaction Processing, and Data

Warehousing. It is also installed with database client Administrator, Runtime, and Custom installations.

- **Oracle Warehouse Builder**: This feature is installed with Oracle Database 11*g*.

Note: With Standard Edition and Enterprise Edition of Oracle Database 11*g* Release 1, Oracle Warehouse Builder with basic features is installed. However, with Enterprise Edition, you can purchase options that extend Oracle Warehouse Builder.

See Also: The following sections and guides for more information:

- Chapter 2, "Oracle Database Preinstallation Requirements" for information about the preinstallation requirements
- *Oracle Database Application Express User's Guide* for more information about Oracle Application Express
- The "Preinstallation Requirements for Oracle Configuration Manager" on page 2-13 for more information
- *Oracle Database Vault Administrator's Guide* for more information about Database Vault
- *Oracle Database Performance Tuning Guide* for more information about Oracle Real Application Testing
- *Oracle Database SQL Developer User's Guide* for more information about Oracle SQL Developer
- *Oracle Warehouse Builder User's Guide* for more information about Oracle Warehouse Builder

Changes in the Install Options

The following are install option changes for Oracle Database 11*g*:

- **Oracle Configuration Manager**: Oracle Configuration Manager is integrated with Oracle Universal Installer. However, it is an optional component for database installation and can be installed with any Oracle Database 11*g* installation.
- **Oracle Data Mining**: Enterprise Edition installation type selects Oracle Data Mining option by default. In Oracle Database 11*g*, the Data Mining metadata is created with `SYS` metadata when you select the **Create Database** option.
- **Oracle Database Vault**: Oracle Database Vault is integrated with Oracle Universal Installer. However, it is an optional component with database installation. To install this product, you have to select the Custom Installation.

  To install Oracle Database Vault with Enterprise Edition, complete the Enterprise Edition installation and then perform a custom installation. Select Oracle Label Security and Oracle Database Vault, and install these products on the same Enterprise Edition database.
- **Oracle HTTP Server**: Starting with Oracle Database 11*g*, Oracle HTTP Server is available on a separate media shipped with Oracle Database. In the previous releases, this product was available as a Companion CD component.

- **Oracle Ultra Search**: Starting Oracle Database 11*g*, Oracle Ultra Search is integrated with Oracle Database. In the previous releases, this product was available as a Companion CD component.
- **Oracle XML DB**: Starting with Oracle Database 11*g*, Oracle XML DB is no longer an optional feature. Database Configuration Assistant installs and configures it for all database installations.

  See Also: The following sections and guides for more information:

  - The "Preinstallation Requirements for Oracle Configuration Manager" on page 2-13 for more information
  - *Oracle Data Mining Administrator's Guide* for more information about Oracle Data Mining
  - *Oracle Database Vault Administrator's Guide* for more information about Oracle Database Vault
  - *Oracle HTTP Server Administrator's Guide* for more information about Oracle HTTP Server
  - *Oracle Ultra Search Administrator's Guide* for more information about Ultra Search
  - *Oracle XML DB Developer's Guide* for more information about Oracle XML DB

Database Configuration Assistant

The following additions and enhancements are made to Database Configuration Assistant:

- Added Support to Configure New Database Options
- Automatic Memory Management
- Oracle Base and Diagnostic Destination Configuration
- Oracle Data Mining
- Secure Database Configuration
- Switching a Database from Database Control to Grid Control Configuration

Added Support to Configure New Database Options

The following options in Oracle Database 11*g* can be configured using Database Configuration Assistant:

- Oracle Application Express
- Oracle Database Vault
- Oracle Warehouse Builder

Automatic Memory Management

This is a new initialization parameter in Oracle Database 11*g* to automate the memory allocation. By default, Database Configuration Assistant now uses `MEMORY_TARGET` instead of specifying individual values for `SGA_TARGET` and `PGA_AGGREGATE_TARGET`. The Memory management page of Database Configuration Assistant has a new option to select automatic memory management.

See Also: The "Using Automatic Memory Management" section of *Oracle Database Administrator's Guide*

Oracle Base and Diagnostic Destination Configuration

The directory that you specify when you are prompted for `ORACLE_BASE` by Oracle Universal Installer is stored in the Oracle home inventory. Database Configuration Assistant uses this value to derive the default database locations and the `DIAGNOSTIC_DEST` parameter. The diagnostic destination location contains all Automatic Diagnostic Repository directories (diagnostic files, such as Alert logs and so on). Starting with Oracle Database Release 11*g*, the initialization parameter settings for background dump, user dump, and core dump destinations are replaced by the Diagnostic Destination.

See Also: "Optimal Flexible Architecture" for more information about Oracle base and diagnostic destination configuration

Oracle Data Mining

In Oracle Database 11*g*, Data Mining metadata is created with the `SYS` metadata. It is created by the `catproc.sql` and other scripts that are run as the `SYS` user. You no longer configure the Data Mining option through the Database Features screen of Oracle Database Configuration Assistant.

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

Secure Database Configuration

Oracle Database 11*g* has new defaults for audit and password profiles. Database Configuration Assistant has a new screen to enable the new security settings during the database creation and existing database configuration.

See Also: The "Database Security Options" on page 3-7 for more information

Switching a Database from Database Control to Grid Control Configuration

In previous releases, Database Configuration Assistant contains the functionality to configure a database either with Database Control, or with Grid Control. You can configure a database either while creating it or later. However, reconfiguring a database from Database Control to Grid Control requires significant manual effort. With Oracle Database 11*g*, Database Configuration Assistant provides the Enterprise Manager Configuration plug-in, which automates the process to switch configuration of a database from Database Control to Grid Control.

See Also: *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for more information about configuring Grid Control

Database Upgrade Assistant

The following additions and enhancements are made to Database Configuration Assistant:

- Command Line Option to Auto Extend System Files
- Express Edition Upgrade
- Integration with Oracle Database 11g Pre-upgrade Tool

Command Line Option to Auto Extend System Files

The command line option AUTOEXTEND facilitates auto extending of the data files as a part of the upgrade. This option automatically extends the data files during the upgrade and turns the autoextend back to its original settings after the upgrade. This option is useful if there is enough space on the disk, and if you do not need to add new data files or manually increase the size of the files.

> **See Also:** The "Altering a Bigfile Tablespace" section in *Oracle Database Administrator's Guide* for more information about the AUTOEXTEND clause

Express Edition Upgrade

For single-instance databases, Oracle Database Upgrade Assistant configuration utility enables you to upgrade from Oracle Database Express Edition (Oracle Database XE) to Oracle Database 11*g*. The XE database files reside under the path ORACLE_BASE/oradata/XE. These files must be copied to a new location as the user may remove the XE Home after upgrade.

Integration with Oracle Database 11*g* Pre-upgrade Tool

Database Upgrade Assistant uses the new pre-upgrade script for Oracle Database 11*g*. This script is used to estimate disk space, initialization parameters, statistics gathering, and providing feedback on possible problem areas.

Moving Data Files into ASM, SAN, and Other File Systems

You can move data files to ASM, OFS, or other storage devices, such as Storage Area Networks (SAN) and Network Area Storage (NAS), as part of the upgrade. If you move the database files during the upgrade, then you can benefit from the typical downtime for this tablespace by rebalancing disks and moving files to a better storage device, such as SAN, NAS, or ASM.

> **See Also:** The Preparing Disk Groups for an Automatic Storage Management Installation on page 2-32 for more information about preparing disk groups for Automatic Storage Management

Oracle Base and Diagnostic Destination Configuration

The directory that you specify when you are prompted for ORACLE_BASE by Oracle Universal Installer is stored in the Oracle home inventory. Database Upgrade Assistant uses this value to derive the default database locations and the DIAGNOSTIC_DEST parameter. The diagnostic destination location contains all ADR directories (diagnostic files, such as the alert logs, trace files, and so on). This diagnostic destination directory is required while upgrading an earlier Oracle Database release to Oracle Database 11*g* release of the database. If the Oracle base directory already exists, then Oracle Database Upgrade Assistant automatically retrieves this information and populates its path. Starting with Oracle Database 11*g* Release 1, the initialization parameter settings for background dump (BACKGROUND_DUMP_DEST), user dump (USER_DUMP_DEST), and core dump (CORE_DUMP_DEST) destinations are replaced by the Diagnostic Destination (DIAGNOSTIC_DEST).

See Also: Appendix D, "Optimal Flexible Architecture"for more information about Oracle base and diagnostic destination configuration

Automatic Storage Management Fast Mirror Resync

Automatic Storage Management fast mirror resync quickly resynchronizes Automatic Storage Management disks within a disk group after transient disk path failures, as long as the disk drive media is not corrupted. Any failures that render a failure group temporarily unavailable are considered transient failures. Disk path malfunctions, such as cable disconnections, host bus adapter or controller failures, or disk power supply interruptions, can cause transient failures. The duration of a fast mirror resync depends on the duration of the outage. The duration of a resynchronization is typically much shorter than the amount of time required to completely rebuild an entire Automatic Storage Management disk group.

See Also: The "Automatic Storage Management Fast Mirror Resync" section in *Oracle Database Storage Administrator's Guide* for more information about ASM fast mirror resync

SYSASM Privilege for Automatic Storage Management Administration

Oracle Database 11*g* introduces an optional system privilege, `SYSASM`, and an optional operating system group, `OSASM`, to secure privileges to perform Automatic Storage Management administration tasks. Oracle recommends that you use `SYSASM` instead of `SYSDBA` for Automatic Storage Management administration, to separate Automatic Storage Management administration from database administration. In a future release, Oracle may restrict access to Automatic Storage Management only to operating system users that are members of the `OSASM` operating system group, and require the use of `SYSASM` to administer Automatic Storage Management.

Note: You can create an operating system group for Automatic Storage Management administrator, in addition to `dba` and `oper` groups.

See Also: The "Authentication for Accessing ASM Instances" section in *Oracle Database Storage Administrator's Guide* for more information about `SYSASM` privilege for Automatic Storage Management

Automatic Maintenance Tasks Management

This feature provides out-of-the-box management of scheduling and resource allocation, such as CPU time, among the various database maintenance tasks, such as Automatic Optimizer Statistics Collection and Automatic Segment Advisor. Maintenance tasks are regulated to the extent that end-user activity gets the necessary resources to finish its work.

See Also: Chapter 24, "Managing Automated Database Maintenance Tasks" of *Oracle Database Administrator's Guide* for more information about Automatic Maintenance task management

Automatic Diagnostic Repository

The Automatic Diagnostic Repository is a feature added to Oracle Database 11*g*. It is a new system managed repository for storing and organizing trace files and other error diagnostic data. The Automatic Diagnostic Repository provides a comprehensive view of the critical errors encountered by the database. This feature also enables you to maintain the relevant data needed for problem diagnostics and their eventual resolution. The Automatic Diagnostic Repository reduces the time to resolve errors and code defects. The repository is stored as a directory structure under the ADR base directory that contains the `diag` directory. The default location of the ADR base directory is set by `DIAGNOSTIC_DEST`. If the `ORACLE_BASE` variable is set, then the default value of `DIAGNOSTIC_DEST` is equal to the value of the `ORACLE_BASE` variable. If the value of the `ORACLE_BASE` variable is not set, then the default value of `DIAGNOSTIC_DEST` is set to `$ORACLE_HOME/log`. However, this location can be changed by using the `DIAGNOSTIC_DEST` parameter of the `init.ora` file.

> **See Also:** The "Automatic Diagnostic Repository (ADR)" section in *Oracle Database Administrator's Guide* for more information about the Automatic Diagnostic Repository

Enhanced Optimal Flexible Architecture

The following enhancements are made to the Optimal Flexible Architecture in Oracle Database 11*g*:

- Oracle Base and Oracle Home
- Flash Recovery Area and Data File Location

Oracle Base and Oracle Home

In Oracle Database 11*g*, Oracle Universal Installer prompts you to specify the Oracle base. You can share this Oracle base across all of the Oracle homes you create on the system. Oracle recommends that you share an Oracle base for all of the Oracle homes created by the same user.

Oracle Universal Installer has a list box where you can edit or select the Oracle base. The installer derives the default Oracle home from the Oracle base location you provide in the list box. However, you can change the default Oracle home by editing the location.

When installing Oracle Clusterware, the Oracle Clusterware home should not be under Oracle base. This is because the `root.sh` script in UNIX operating systems changes the ownership of the parent directories up to the root file system (`/`) to the `root` user. If you specify an Oracle Clusterware home under Oracle base, Oracle Universal Installer displays an error.

The following are the changes made in Oracle Database 11*g* with respect to Oracle base to make it Optimal Flexible Architecture compliant:

- `ORACLE_BASE` is a recommended environment variable. However, this variable will be made mandatory in future releases.
- Oracle recommends that you create the flash recovery area and data file location under Oracle base.

> **See Also:** *Oracle Clusterware Installation Guide for Linux* for more information about Oracle Clusterware home

Flash Recovery Area and Data File Location

In Oracle Database 10*g*, the default locations for the flash recovery area and data files are one level above the Oracle home directory. However, in Oracle database 11*g*, Oracle base is the starting point to set the default locations for flash recovery and data files. However, Oracle recommends that you keep the flash recovery area and data file location on separate disks. To mount the disks you can use the following mount points for flash recovery area and data file location respectively:

```
$ORACLE_BASE/flash_recovery_area
$ORACLE_BASE/oradata
```

Oracle recommends you use separate disks for `oradata`, flash recovery, and the Oracle home.

If you install Oracle RAC, then you must share flash recovery area and data file location among all the nodes.

> **See Also:** Appendix D, "Optimal Flexible Architecture" for more information about Optimal Flexible Architecture

Oracle Direct Network File System Client

This feature is implemented as a Direct Network File System (NFS) client as a part of Oracle RDBMS Kernel in Oracle Disk Manager library. NAS-based storage systems use NFS to access data. In Oracle Database 10*g*, NAS storage devices are accessed using the operating system provided kernel NFS driver, which require specific configuration settings to ensure its efficient and correct usage with Oracle. The following are the major problems that arise in correctly specifying configuration parameters:

- NFS clients are very inconsistent across platforms and vary across operating system releases.
- The configuration parameters are difficult to tune. There are more than 20 NFS parameters and they have subtle differences across platforms.
- NFS client stack is designed for general purpose. Therefore, it contains features like file attribute management that are not required for Oracle.
- Oracle Direct NFS implements NFS version 3 protocol within the Oracle RDBMS kernel.

The following are the main advantages of implementing Oracle Direct NFS client functionality in Oracle RDBMS Kernel:

- It enables complete control over input-output paths to network file servers, resulting in predictable performance, simplified configuration management, and superior diagnostics.
- Its operations avoid the kernel NFS layer bottlenecks and resource limitations. However, the kernel is still used for network communication modules.
- It provides a common NFS interface for Oracle for potential use on all host platforms and supported NFS servers.
- It enables improved performance through load balancing across multiple connections to NFS servers and deep pipelines of asynchronous input-output operations with improved concurrency.

> **See Also:** *Oracle Clusterware Installation Guide for Linux* for more information in Network File System

Deprecated Components in Oracle Database 11*g* Release 1 (11.1)

The following is a list of components that were part of Oracle Database 10*g* release 2 (10.2), and are not available for installation with Oracle Database 11*g*:

- *i*SQL*Plus
- Oracle Workflow
- Oracle Data Mining Scoring Engine
- Oracle Enterprise Manager Java console

1

Overview of Oracle Database Installation

This chapter describes the different installation types of Oracle Database for Linux and issues to consider before you install Oracle Database:

- Planning the Installation
- Installing the Linux Operating System
- New Oracle Products Installed with This Release
- Installation Considerations
- Oracle Database Installation Types
- Database Configuration Options
- Database Storage Options
- Database Management Options
- Database Backup and Recovery Options
- E-mail Notification Options

Planning the Installation

The Oracle Database installation process consists of the following phases:

1. **Read the release notes:** Read *Oracle Database Release Notes for Linux* 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. **Review the licensing information:** Although the installation media in the media pack contain many Oracle components, you are permitted to use only those components for which you have purchased licenses.

   Oracle Support Services does not provide support for components for which licenses have not been purchased.

   See Also: *Oracle Database Licensing Information* for more details

3. **Plan 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 H, 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 connections.

> **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.

4. **Complete preinstallation tasks:** Chapter 2 describes preinstallation tasks that you must complete before installing the product.
5. **Install the software:** Use the following sections to install Oracle Database:
   - Chapter 3 describes how to use Oracle Universal Installer to install Oracle Database and Automatic Storage Management.
   - Appendix A provides information on performing noninteractive (silent) installations, which you may want to use if you need to perform multiple installations of Oracle Database.
   - Appendix B provides information on cloning Oracle home.
   - Appendix F describes globalization support information.
   - Appendix G provides troubleshooting advice in case you encounter problems with the installation.
   - Chapter 6 describes how to remove Oracle Database.
6. **Complete postinstallation tasks:** Chapter 4 describes recommended and required postinstallation tasks.
7. **Get started using Oracle Database:** Use the following sections to get started with 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 network attached storage devices, which you can use to store Oracle database files and Oracle software.
   - Appendix D describes the Optimal Flexible Architecture, which is a set of guidelines that ensures reliable Oracle installations that requires little maintenance.
   - Appendix E explains the method to manage Oracle Database port numbers.

Installing the Linux Operating System

This section provides information about installing a supported Linux distribution. It contains the following topics:

- Completing a Minimal Linux Installation
- Completing a Default Linux Installation
- About the Oracle Validated Configuration RPM
- Installing the Oracle Validated Configuration RPM

Completing a Minimal Linux Installation

To complete a minimal Linux installation, select a minimal install option (either a custom installation where you select the Minimal option from Package Group Selection, or where you deselect all packages except for the Base pack). This installation lacks many RPMs required for installation. However, when you install the Oracle Validated RPM for your platform, the RPM downloads the minimum number of packages required to run Oracle Clusterware and Oracle Database.

Unbreakable Linux Network (ULN) customers can obtain the Oracle Validated RPM by using up2date. If you are not a ULN customer, and you are running Red Hat or Oracle Enterprise Linux, then you can obtain the Oracle Validated RPM at the following URLs:

Enterprise Linux 4:

`http://oss.oracle.com/el4/oracle-validated/`

Enterprise Linux 5:

`http://oss.oracle.com/el5/oracle-validated/`

If you are not a member of ULN or RHN (Red Hat support network) and you are an Oracle support customer, then you can download instructions to configure a script that replicates Oracle Validated RPM package downloads at the following URL:

`https://metalink.oracle.com`

Search for "minimal Linux".

Note: The Oracle Validated RPM installs the X11 client libraries, but does not install the X Window System server packages. To use graphic user interfaces such as Oracle Universal Installer, configuration assistants, and Enterprise Manager, set the display to a system with X Window System server packages.

Completing a Default Linux Installation

If you do not install the Oracle Validated RPM, then Oracle recommends that you install your Linux operating system with the default software packages (RPMs). This installation includes most of the required packages and helps you limit manual checks of package dependencies. Oracle recommends that you do not customize the RPMs during installation.

For information about a default installation, log on to OracleMetalink:

`https://metalink.oracle.com`

Search for "Default RPM."

After installation, review system requirements for your distribution to ensure that you have all required kernel packages installed, and complete all other configuration tasks required for your distribution, and for your system configuration.

About the Oracle Validated Configuration RPM

If your Linux distribution is Oracle Enterprise Linux, or Red Hat Enterprise Linux, and you are an Unbreakable Linux Network (ULN) or Red Hat network customer, then you can complete most preinstallation configuration tasks by using the Oracle Validated Configurations Setup RPM, available from ULN, or from the Oracle Open

Source Software URLs, as described in "Completing a Minimal Linux Installation" on page 1-3.

When it is installed, the Oracle Validated Configuration RPM sets and verifies system parameters based on recommendations from the Oracle Validated Configurations program, and installs any additional packages needed for installing Oracle Clusterware and Oracle Database. It creates an oracle software owner (`oracle`), and the OSDBA group (`dba`) and Oracle Inventory group (`oinstall`). It also updates `sysctl.conf` settings, system startup parameters, user limits, and driver parameters to values tested for performance.

To become a ULN customer, contact your sales representative, or purchase a license from the Unbreakable Linux store:

`http://oraclestore.oracle.com/linux`

To register your server on ULN, or to find out more information about ULN, refer to the following URL:

`https://linux.oracle.com`

Installing the Oracle Validated Configuration RPM

Use the following procedure to subscribe to Oracle Unbreakable Linux channels, and to add the Oracle Software for Enterprise Linux channel that distributes the Oracle Validated Configurations Setup RPM:

1. Complete a default Oracle Enterprise Linux workstation installation, or a default Red Hat Enterprise Linux installation.
2. Register your server with Unbreakable Linux Network (ULN). By default, you are registered for the Enterprise Linux Latest channel for your operating system and hardware.
3. Log in to ULN at the following URL:

   `https://linux.oracle.com`
4. Click the **Systems** tab, and in the System Profiles list, select a registered server. The System Details window opens, and displays the subscriptions for the server.
5. From the Available Channels list, select the Oracle Software for Enterprise Linux channel that is appropriate for your installation of Linux (for example: "Oracle Software for Enterprise Linux 4 (x86_64)."
6. Click Subscribe.
7. From a terminal session, as root, enter the following command:

   ```
   # up2date --nox --show-channels
   ```

   You should see output indicating that you have subscribed to the Oracle Software for Enterprise Linux channel. For example:

   ```
   el4_i386_latest
   el4_i386_oracle
   ```
8. Open a terminal session as root, and install the Oracle Validated Configurations Setup RPM with `up2date`, using the following command:

   ```
   # up2date --install oracle-validated
   ```
9. Repeat steps 1 through 8 on all other servers in your cluster.

Note: Check the Oracle Validated Configuration RPM log file to review system configuration changes:

```
/etc/sysconfig/oracle-validated/results/orakernel.log
```

New Oracle Products Installed with This Release

The following products are installed by default when installing Oracle Database 11*g* Release 1:

- Oracle Application Express
- Oracle Warehouse Builder
- Oracle Configuration Manager
- Oracle Database Vault
- Oracle SQL Developer

Oracle Application Express

Oracle Application Express is a tool for development and deployment of Web applications for an Oracle database. It improves the productivity, security, reliability, and performance of Oracle database. With little programming or scripting and only a Web browser, you can build reporting and data entry applications on existing tables, views, or data imported from spreadsheets.

Oracle Warehouse Builder

Oracle Warehouse Builder is the only enterprise business intelligence integration design tool that manages the full life-cycle of data and metadata for the Oracle Database. It provides an easy to use graphical environment to rapidly design, deploy, and manage business intelligence systems.

With the Standard and Enterprise Editions of Oracle Database, you can use Oracle Warehouse Builder that enables you to integrate and transform data into high quality information. When you install the Standard Edition or Enterprise Edition of Oracle Database, the installation provides you with components necessary for Oracle Warehouse Builder, including an unpopulated schema, `OWB_SYS`. Unlock the `OWB_SYS` schema and install the Oracle Warehouse Builder software on a client computer, as described in *Oracle Warehouse Builder Installation and Administration Guide*.

Oracle Configuration Manager

Oracle Configuration Manager is a utility that can be optionally configured when installing the Oracle Database. Oracle Configuration Manager is used to collect and upload the configuration information to the Oracle configuration repository.

The following are some of the benefits of Oracle Configuration Manager:

- Reduces time for resolution of support issues
- Provides pro-active problem avoidance
- Improves access to best practices and the Oracle knowledge base
- Improves understanding of customer's business needs and provides consistent responses and services

Oracle Configuration Manager can now be installed in two modes:

- **Connected Mode**: This mode is recommended if your server has direct connection to the Internet or connection through a proxy server. In this mode, configuration data is automatically collected and uploaded to the Oracle system. Updates to Oracle Configuration Manager occur automatically.
- **Disconnected Mode**: This mode is recommended if your server does not have a connection to Internet. In this mode, you can collect configuration data manually by using the emCCR collect command. When you run this command, the collected configuration data is stored in the `$ORACLE_HOME/ccr/state/upload/ocmconfig.jar` file. You can then upload this file to the Oracle server.

Oracle Database Vault

Oracle Database Vault enables you to secure business data in ways that were not possible before. Database Vault uses a multifactored and multilayered approach to implementing database security. Before you plan the upgrade process, become familiar with the features of Oracle Database Vault. The *Oracle Database Vault Administrator's Guide* discusses the basic features of Oracle Database Vault.

Note: You cannot remove or uninstall the Database Vault. However, you can disable Oracle Database Vault. Refer to the *Disable Oracle Database Vault* section in the *Oracle Database Upgrade Guide* guide for more details.

Oracle SQL Developer

Oracle SQL Developer is a graphical version of SQL*Plus that gives database developers a convenient way to perform basic tasks. Following are the functions you can perform with Oracle SQL Developer:

- Browse, create, edit, and delete (drop) database objects
- Run SQL statements and scripts
- Create, edit, compile and debug PL/SQL code
- Create, edit and update data
- Import data, export data and data definition language
- View and create reports
- View metadata and data of Microsoft Access, Microsoft SQL Server, and MySQL databases

Installation Considerations

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

- Hardware and Software Certification
- Multiple Oracle Homes Support
- Oracle Cluster Synchronization Services
- Using Network Attached Storage or NFS File Systems

- Default Audit Policy and Initialization Parameters

Hardware and Software Certification

The platform-specific hardware and software requirements included in this guide were current when 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:

`https://metalink.oracle.com`

You must register online before using Oracle*MetaLink*. After logging in, click **Certify** on the top right-hand side of the screen. The Certifications page appears. You can choose the certification either by product or by platform.

Other options include Oracle's Certification Matrices, Desupport Notices, and Product Availability.

Third-Party Database Certification for SQL Developer

SQL Developer can be used to view metadata and data of several non-Oracle databases. The following table lists the third-party database certifications.

Database	Releases	Notes
Microsoft Access	Access 97 Access 2000 Access 2003	For any Access release: no JDBC driver needed, but you must ensure read access to the system tables in the `.mdb` file.
Microsoft SQL Server	SQL Server 7 SQL Server 2000 SQL Server 2005	For any Microsoft SQL Server release: JDBC driver `jtds-1.2.2.jar` required. This is included in the `jtds-1.2-dist.zip` file available from `sourceforge.net`.
MySQL	MySQL 3.x MySQL 4.x MySQL 5.x	For any MySQL release: JDBC driver required. For MySQL 5.x: `mysql-connector-java-5.0.4-bin.jar` is required, which is included in `mysql-connector-java-5.0.4.zip`.

Multiple Oracle Homes Support

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

Installing the Software on a System with an Existing Oracle Installation

You must install this product into a new Oracle home directory. You cannot install products from one release of Oracle Database into an Oracle home directory of a different release. For example, you cannot install Oracle Database 11g Release 1 software into an existing Oracle9*i* 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 more than once on the same system if each installation is installed in a separate Oracle home directory.

Oracle Cluster Synchronization Services

The first time you install Oracle Database 11*g* and choose Automatic Storage Management as the storage option on a system, the installation configures and starts a single-node version of the Oracle Cluster Synchronization Services (CSS) daemon. The CSS daemon is required to enable synchronization between an Automatic Storage Management instance and the database instances that rely on it for database file storage. By default, Oracle Universal Installer does not configure CSS. Oracle Universal Installer configures these services only if you choose Automatic Storage Management as a storage or recovery option. Because the daemon must be running before any Automatic Storage Management instance or database instance starts, it is configured to start automatically when the system boots.

For Oracle RAC installations, the CSS daemon is installed with Oracle Clusterware in a separate Oracle home directory (also called the Clusterware home directory). For single-node installations, the CSS daemon is installed in and runs from the same Oracle Database home that runs Automatic Storage Management.

If you have installed CSS from the same Oracle home as Oracle Database, then use caution when removing Oracle Database software from the system. Before you remove an Oracle home directory that contains Oracle Database 11*g*, you must either delete the CSS daemon configuration, or if necessary, reconfigure the CSS daemon to run from another Oracle home directory.

Note: If you plan to have more than one Oracle Database 11*g* installation on a single system and you want to use Automatic Storage Management for database file storage, then Oracle recommends that you run the CSS daemon and the Automatic Storage Management instance from the same Oracle home directory and use different Oracle home directories for the database instances.

See Also:

- "Automatic Storage Management" on page 1-13
- "Reconfiguring Oracle Cluster Synchronization Services" on page 6-5
- "Removing Oracle Software" on page 6-8

Using Network Attached Storage or NFS File Systems

Oracle Database 11*g* must be able to verify that writes to a disk are completed successfully. NFS file systems, including file systems on NAS devices, may not be able to guarantee that writes to a disk are completed successfully, and this may lead to possible data file corruption. Oracle recommends that you do not store files on NFS mounted file systems unless the storage vendor and storage device are listed in the Oracle Storage Compatibility Program list.

If a storage device is supported, then you can use it to store Oracle software files, Oracle database files, or both.

See Also: Appendix C for guidelines about using NFS and NAS devices for Oracle software or database files

Default Audit Policy and Initialization Parameters

Oracle Database Vault installs a baseline database auditing policy. This policy covers the access control configuration information stored in Database Vault database tables, information stored in Oracle Catalog (rollback segments, tablespaces, and so on), the use of system privileges, and Oracle Label Security configuration. When you install Oracle Database Vault, the security specific database initialization parameters are initialized with default values.

> **See Also:** *Oracle Database Vault Administrator's Guide* for more information on the database audit policy

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 11*g* Release 1 (11.1) on Linux, Oracle Universal Installer provides two interactive methods that you can use to install Oracle Database:

- Basic: Select this installation method if you want 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: Select this installation method if you want to complete any of the following tasks:
  - Upgrade an existing database
  - 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
  - Configure Automatic Storage Management for database storage
  - Specify different passwords for administrative schemas
  - Configure automated backups or Oracle Enterprise Manager notifications
  - Configure Oracle Configuration Manager
  - 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**.

See Also: "Reviewing Installation Guidelines" on page 3-1 for additional information on Oracle database installation

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 need to perform multiple installations on similarly configured systems or if the system where you want to install the software does not have X Window system software installed.

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

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

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

Oracle Database Installation Types

You can choose to install the Oracle Client separately. You cannot install Oracle Database Client during an Oracle Database Installation. You can choose one of the following installation types when installing Oracle Database 11*g*:

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.

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

See Also:

- *Oracle Database Client Installation Guide for Linux* for Oracle Database Client installation instructions
- *Oracle Database Licensing Information* for more information about the features available with each Oracle Database edition and for information about licensing

Note: The installation process is the same for all the installation types. Ensure that you install the licensed editions.

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 Oracle Database Configuration Assistant to create it. You can choose to create one of the preconfigured database types, which are designed for a variety of different applications, modify one of the preconfigured database types, or create a customized database to suit the 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 Oracle Database Configuration Assistant for a description of these preconfigured database types.

Installation Choices that Affect Database Creation

Oracle Universal Installer runs Oracle 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 Oracle Database Configuration Assistant in noninteractive mode to create the database after it installs the software.

  Note: Oracle recommends that you use this method to create a database if you have not previously created one.

- 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 Oracle Database Configuration Assistant in interactive mode. Using the screens in Oracle Database Configuration Assistant, you can either modify one of the preconfigured

database types 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 Oracle 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 you can use Oracle Database Configuration Assistant to create one after you have installed the software. For more information about using Oracle Database Configuration Assistant to create a database after installation, refer to the *Oracle Database 2 Day DBA* manual.

Database Storage Options

If you choose to create a database during the installation, you can specify one of the following storage options for database files:

- File System
- Automatic Storage Management
- Raw Devices

File System

If you choose the file system option, then Oracle 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 D and distribute the database files over more than one disk.

- 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 more than one file system mount point for database storage.

- A network file system (NFS) mounted from a certified network attached storage (NAS) device

  If the NAS device is certified by Oracle, then you can store the database files on them.

  See Also: The "Using Network Attached Storage or NFS File Systems" section on page 1-8 for more information about certified NAS and NFS devices

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 need to 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

Automatic Storage Management

Automatic Storage Management is a high-performance storage management solution for Oracle Database files. It simplifies the management of a dynamic database environment, such as creating and laying out databases and managing disk space.

Automatic Storage Management can be used with single database installations, multiple database installations, and in Oracle RAC environments. It can be used with databases created in Oracle Database 10*g* Release 1 (10.1.0.3 or later). However, Oracle Database 11*g* Release 1 (11.1) databases can use Automatic Storage Management from Oracle Database 10*g* Release 1 (10.1). If a site has multiple single-instance databases, then you can use Oracle Clusterware to consolidate multiple databases into a single clustered pool of storage managed by Automatic Storage Management. Automatic Storage Management manages the storage of all database files, such as redo logs, control files, and data pump export files. However, it does not manage Oracle Database executable binary files.

At a high level, implementing Automatic Storage Management involves allocating partitioned disks for Oracle Database with preferences for striping and mirroring. Automatic Storage Management manages the disk space for you. This helps avoid the need for traditional disk management tools, such as Logical Volume Managers (LVM), file systems, and the numerous commands necessary to manage both. The synchronization between Automatic Storage Management and the database instance is handled by CSS.

The following are components of an Automatic Storage Management installation:

- Automatic Storage Management Disk Groups
- Automatic Storage Management Instance
- General Steps for Installing Automatic Storage Management

Automatic Storage Management Disk Groups

A disk group is a set of disk devices that Automatic Storage Management manages as a single unit. Each disk device can be an individual physical disk, a multiple disk device, such as a RAID storage array or logical volume, or even a partition on a physical disk. However, in most cases, disk groups consist of one or more individual physical disks. To enable Automatic Storage Management to balance input-output operation and storage efficiently within the disk group, you must ensure that all devices in the disk group have similar, if not identical, storage capacity and performance.

You can set the redundancy and striping attributes of individual file types within a disk group by using Automatic Storage Management disk group templates. When you create a disk group, Automatic Storage Management creates a set of default templates for that disk group. Default template settings depend on the disk group type. For example, the default template for control files for a normal redundancy disk group sets three-way mirroring. All other file templates are two-way mirrored. For a high redundancy disk group, the default mirroring cannot be changed, which implies that all files are always three-way mirrored in a high redundancy disk group. You can

modify the default templates to suit your site's needs. Refer to *Oracle Database Administrator's Guide* for more information.

Automatic Storage Management spreads data evenly across all the devices in the disk group to optimize performance and utilization. You can add or remove disk devices from a disk group without shutting down the database. When you add or remove disks, Automatic Storage Management rebalances the files across the disk group. You can create multiple disk groups to handle specific tasks, such as backup and recovery operations, in addition to regular file storage activities.

When you add a device to a disk group, you can specify a failure group for that device. Failure groups identify disk devices that have common failure characteristics, for example, devices that are attached to the same controller. If the controller fails, then all devices attached to it become unavailable. By default, each device also belongs to its own failure group. By using the failure groups you specify, Automatic Storage Management can distribute data among the devices in the disk group to minimize the risk of data loss caused by component failures.

Automatic Storage Management Instance

The Automatic Storage Management instance manages Automatic Storage Management disk groups. This instance must be running before you can start a database instance that uses Automatic Storage Management. When you choose Automatic Storage Management as a database storage mechanism, this instance is created and started, if necessary. For a single-instance Oracle Database installation, you only need one Automatic Storage Management instance, regardless of the number of database instances on the system. The Automatic Storage Management instance on any node in a single cluster can handle any combination of disk group types.

General Steps for Installing Automatic Storage Management

To install Automatic Storage Management, you use Oracle Universal Installer. This installation guide provides the following general steps for installing Automatic Storage Management:

See Also:

- "Oracle Cluster Synchronization Services" on page 1-8 for information regarding configuration of CSS daemon on a single node.
- *Oracle Database 2 Day DBA* for a general overview, from a non-platform perspective, of Automatic Storage Management
- *Oracle Database New Features Guide* for information about new features in this release of Automatic Storage Management
- *Oracle Database Administrator's Guide* for a more detailed description of Automatic Storage Management
- The Oracle Technology Network Web site, for additional information on Automatic Storage Management, which you can visit at

  `http://www.oracle.com/technology/products/database/asm`

1. Determine the disk requirements of the site and, if necessary, create one or more disk partitions for Automatic Storage Management.

"Preparing Disk Groups for an Automatic Storage Management Installation" on page 2-32 provides guidelines on how to determine the disk requirements of the site.

2. Run Oracle Universal Installer to install and create an Automatic Storage Management instance and to create Automatic Storage Management disk groups.

   "Step 1: Reviewing Automatic Storage Management Installation Considerations" on page 3-15 provides guidelines on where to install Automatic Storage Management and other installation considerations.

   "Step 2: Installing the Automatic Storage Management Instance and configuring Disk Groups" on page 3-15 describes how to create an Automatic Storage Management instance and disk groups.

   After you create an Automatic Storage Management instance and its associated disk groups, subsequent databases that you create can use Automatic Storage Management for file storage management. If you have databases that were created before you installed Automatic Storage Management, then you can migrate them to Automatic Storage Management by using the Enterprise Manager Migrate Database Wizard. This wizard is available in Oracle Enterprise Manager Grid Control or Oracle Enterprise Manager Database Control. Alternatively, you can use Oracle Database Recovery Manager (RMAN) to perform the migration.

3. To create an Oracle Database with Automatic Storage Management refer "Step 3: Installing Oracle Database to Use Automatic Storage Management" on page 3-18

4. Test the Automatic Storage Management installation.

   "Step 3: Installing Oracle Database to Use Automatic Storage Management" on page 3-18 provides a simple test you can try to check that the Automatic Storage Management installation was successful. "Managing Automatic Storage Management" on page 5-3 explains how to start and access Automatic Storage Management and which Oracle Database tools you can use to manage it.

Raw Devices

Raw devices are disk partitions or logical volumes that have not been formatted with a file system. Raw devices are not available as a storage option when creating a database as a part of a new Oracle Database 11*g* installation. However, upgrading the existing version of raw devices are supported. Raw device is available as a storage option when you run Oracle Database Configuration Assistant from the following location:

```
$ORACLE_HOME/bin/dbca
```

When you use raw devices for database file storage, Oracle writes data directly to the partition or volume, bypassing the operating system file system layer. However, since raw devices can be difficult to create and administer, and because the performance gains over modern file systems are minimal, Oracle recommends that you choose Automatic Storage Management or file system storage in preference to raw devices.

Database Management Options

To simplify database administration, Oracle provides a Web-based management tool called Oracle Enterprise Manager. There are two ways that you can deploy Oracle Enterprise Manager:

- Deploy Oracle Enterprise Manager 10g centrally in the environment

To deploy Oracle Enterprise Manager centrally, you must install at least one Oracle Management Repository and one Oracle Management Service within the environment, then install an Oracle Enterprise Management Agent on every computer that you want to manage. You can then use a single HTML interface to manage and monitor software and hardware targets on all of those systems. Targets can include Oracle databases, application servers, Net listeners, and third-party software. This single interface is called Oracle Enterprise Manager Grid Control (or simply Grid Control).

Note: Oracle Enterprise Manager 10*g* is available separately on the Oracle Enterprise Manager Grid Control installation media.

- Deploy Oracle Enterprise Manager Database Control locally on the database system

  Oracle Enterprise Manager Database Control software is installed by default with every Oracle Database installation except Custom. During a Custom installation, you can choose not to install Oracle Enterprise Manager Database Control. However, Oracle recommends that you do install it. This local installation provides a Web-based interface called Oracle Enterprise Manager Database Control. The Database Control is similar in function to the Grid Control, but it can manage only a single database. If you want to administer more than one database on this system, then you must either configure a separate Database Control for each database, or install Oracle Enterprise Manager 10*g* Grid Control.

Note: Refer to the *Oracle Enterprise Manager Concepts* manual and the *Oracle Enterprise Manager Grid Control Installation and Basic Configuration* manual for more information Oracle Enterprise Manager 10*g*.

This section contains the following topics:

Management Options for Preconfigured Databases

When you choose to create a preconfigured database during the installation, you must select the Oracle Enterprise Manager interface that you want to use to manage the database. The following options are available:

- Use Grid Control for central database management

  This option is available only if an Oracle Management Agent is installed on the system. When Oracle Universal Installer detects an Oracle Management Agent on the system, you can choose this option and specify the Oracle Management Service that you want to use to manage the database.

  If an Oracle Management Agent is not installed, then you need to use Database Control to manage the database. However, if Oracle Management Agent is installed after Oracle Database, then you can use Grid Control to manage this database.

- Use Database Control for local database management

This option is selected by default if an Oracle Management Agent is not installed on the system. However, even if a Management Agent is installed, you can still choose to configure Database Control to manage the database.

Management Options for Custom Databases

If you choose the Advanced database configuration option or choose to create a database during a Custom installation, then Oracle Universal Installer runs Oracle Database Configuration Assistant in interactive mode. Using a screen in Oracle Database Configuration Assistant, you can specify the Oracle Enterprise Manager interface that you want to use to manage the database. Alternatively, you can also choose not to configure the database with Enterprise Manager.

Oracle recommends that you configure the database to use Enterprise Manager during installation. However, if you choose not to configure the database to use Enterprise Manager during the installation, then you can use Oracle Database Configuration Assistant after the installation to configure the database to use it.

Features Provided by Oracle Enterprise Manager Database Control

Oracle Enterprise Manager Database Control provides a Web-based user interface that enables you to monitor, administer, and maintain an Oracle database. You can use it to perform all of the database administration tasks. You can also use it to determine information about the database, such as:

- Instance name, database version, Oracle home location, media recovery options, and other instance data
- Current instance availability
- Database alert information
- Session and SQL-related performance information
- Space usage matrix

In addition, it provides you with automatic notification of security alerts and it provides the ability to download and apply patches for the software.

Database Backup and Recovery Options

If you choose to use Oracle Enterprise Manager Database Control during the installation, then you can optionally enable automated database backups that use the Oracle-suggested default backup strategy. You do not have to enable automated backups during the installation. If you prefer, you can use Oracle Enterprise Manager Database Control or Grid Control to configure automated backups after you install the software and create a database.

This section contains the following topics:

- Enabling Automated Backups
- Backup Job Default Settings

See Also:

- *Oracle Database 2 Day DBA* for information about using Oracle Enterprise Manager Database Control to configure or customize automated backups or to recover a backed up database
- *Oracle Database Backup and Recovery User's Guide* for more detailed information about defining a backup strategy and backing up and recovering Oracle databases

Enabling Automated Backups

If you enable automated backups, then Oracle Enterprise Manager schedules a daily backup job that uses Oracle Recovery Manager (RMAN) to back up all of the database files to an on disk storage area called the flash recovery area. The first time the backup job runs, it creates a full backup of the database. Subsequent backup jobs perform incremental backups, which enable you to recover the database to its state at any point during the preceding 24 hours.

To enable automated backup jobs during installation, you must specify the following information:

- The location of the flash recovery area

  You can choose to use either a file system directory or an Automatic Storage Management disk group for the flash recovery area. To set the default values for flash recovery area and data file location, use Oracle base as the starting point.

  - Default Flash recovery area: `$ORACLE_BASE/flash_recovery_area`
  - Default data file location: `$ORACLE_BASE/oradata`

  The default disk quota configured for the flash recovery area is 2 GB. For Automatic Storage Management disk groups, the required disk space depends on the redundancy level of the disk group that you choose. Chapter 2 describes how to choose the location of the flash recovery area and identifies its disk space requirements.

- An operating system user name and password for the backup job

  Oracle Enterprise Manager uses the operating system credentials that you specify when running the backup job. The user name that you specify must belong to the UNIX group that identifies database administrators (the `ORA_DBA` group). This user also must have Logon As A Batch Job privilege.

Backup Job Default Settings

If you enable automated backups after choosing one of the preconfigured databases during the installation, then automated backup is configured with the following default settings:

- The backup job is scheduled to run nightly at 2 a.m.
- The disk quota for the flash recovery area is 2 GB.

If you enable automated backups by using Oracle Database Configuration Assistant, either during or after the installation, then you can specify a different start time for the backup job and a different disk quota for the flash recovery area.

E-mail Notification Options

If you choose to use the Oracle Enterprise Manager Database Control during the installation, then you can configure Enterprise Manager to send e-mail when specific events occur. These events can include occurrences such as disk space reaching a critical limit (a threshold), or a database shutting down unexpectedly.

If you choose to enable e-mail notifications, then you must specify the following information:

- The host name of a Simple Mail Transfer Protocol (SMTP) server
- The e-mail address that should receive the alerts

  The e-mail address that you specify could belong to an individual or it could be a shared e-mail account or a distribution list.

You can use Enterprise Manager Database Control to set up, change, or customize e-mail notifications after you have created the database.

Migration Consideration

Oracle Database 11g Release 1 (11.1) database for 32-bit Linux can be migrated to an Oracle Database 11g Release 1 (11.1) database for 64-bit Linux. Refer to "Database Migration from a 32-Bit Linux to 64-Bit Linux Computer" section in the *Oracle Database Administrator's Reference for Linux and UNIX* for migration information.

Upgrade Considerations

For information about upgrading a earlier release of Oracle Database to Oracle Database 11*g* Release 1 (11.1), refer to *Oracle Database Upgrade Guide*. The following sections provide additional platform-specific upgrade information that you should review before upgrading an existing database:

AL24UTFFSS Character Set

Note: The information in this section does *not* apply to an upgrade of a eelease 1 (9.0.1) or later release of Oracle Database.

Before you upgrade an existing database that uses the AL24UTFFSS character set, you must upgrade the database character set to UTF8. Oracle recommends that you use the Character Set Scanner (`csscan`) utility for data analysis before attempting to upgrade the existing database character set.

The Character Set Scanner utility checks all character data in the database and tests for the effects of, and problems with, changing the character set encoding. Before running the Character Set Scanner utility, set the shared library path environment variable for the platform to include the `$ORACLE_HOME/lib` directory. The shared library path environment path that you need to set is `LD_LIBRARY_PATH`.

Note: AL32UTF8 is the Oracle Database character set that is appropriate for XMLType data. It is equivalent to the IANA registered standard UTF-8 encoding, which supports all valid XML characters.

Do not confuse Oracle Database database character set UTF8 (no hyphen) with database character set AL32UTF8 or with character encoding UTF-8. Database character set UTF8 has been superseded by AL32UTF8. Do not use UTF8 for XML data. UTF8 supports only Unicode version 3.0 and earlier; it does not support all valid XML characters. AL32UTF8 has no such limitation.

Using database character set UTF8 for XML data could potentially cause a irrecoverable error or affect security negatively. If a character that is not supported by the database character set appears in an input-document element name, then a replacement character (usually "?") is substituted for it. This will terminate parsing and raise an exception.

See Also: *Oracle Database Globalization Support Guide* for more information about Character Set Support

Upgrading an Oracle Database Installed on Red Hat Enterprise Linux 2.1

If you have the 8.1.7, 9.0.1, 9.2.0, or 10.1 release of Oracle Database installed on Red Hat Enterprise Linux 2.1, then you must first upgrade the operating system to Red Hat Enterprise Linux 3 (update 4) before you upgrade the database. To do this, perform any one of the following procedures:

See Also: *Oracle Database Upgrade Guide*

- Upgrade the operating system. Then, upgrade the database either manually or by using Oracle Database Upgrade Assistant. The detailed information on preserving database environment while upgrading the operating system is available at the following URL:

  http://www.oracle.com/technology/tech/linux/pdf/rhel_23_upgrade.pdf

- Copy the database files. This procedure involves the following steps:

  1. Copy the database files from the computer running Red Hat Enterprise Linux 2.1 to the one running Red Hat Enterprise Linux 3.0.
  2. Re-create the control files on the computer running Red Hat Enterprise Linux 3.0.
  3. Manually upgrade the database.

  Note: You cannot use Oracle Database Upgrade Assistant if you follow this method. However, this method lets you easily revert to the earlier database.

- Upgrade the database by using the Export/Import utilities.

2

Oracle Database Preinstallation Requirements

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

> **Note:** This guide contains information required to install Oracle Database 11*g* Release 1 (11.1) on various platforms of Linux. Ensure that you review information related to the platform on which you intend to install Oracle Database 11*g*.

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 one of the following procedures:

Note: Unless you intend to complete a silent-mode installation, you must install the software from an X Window System workstation, an X terminal, or a PC or other system with X server software installed.

For more information about silent-mode installations, refer to Appendix A.

- Following are the steps for installing the software from an X Window System workstation or X terminal:

  1. Start a local terminal session, for example, an X terminal (xterm).
  2. If you are not installing the software on the local system, then enter the following command to enable the remote host to display X applications on the local X server:

     ```
     $ xhost fully_qualified_remote_host_name
     ```

     For example:

     ```
     $ xhost somehost.us.example.com
     ```

  3. If you are not installing the software on the local system, then use the ssh, rlogin, or telnet command to connect to the system where you want to install the software:

     ```
     $ telnet fully_qualified_remote_host_name
     ```

  4. If you are not logged in as the root user, then enter the following command to switch user to root:

     ```
     $ sudo sh
     password:
     #
     ```

- Following are the steps for installing the software from a PC or other system with X server software:

  Note: If necessary, refer to the X server documentation for more information about completing this procedure. Depending on the X server software that you are using, you may need to complete the tasks in a different order.

  1. Start the X server software.
  2. Configure the security settings of the X server software to permit remote hosts to display X applications on the local system.
  3. Connect to the remote system where you want to install the software and start a terminal session on that system, for example, an X terminal (xterm).
  4. If you are not logged in as the root user on the remote system, then enter the following command to switch user to root:

     ```
     $ sudo sh
     password:
     #
     ```

Checking the Hardware Requirements

The system must meet the following minimum hardware requirements:

- Memory Requirements
- System Architecture
- Disk Space Requirements
- Recommended Hardware Requirement for SQL Developer

Memory Requirements

The following are the memory requirements for installing Oracle Database 11*g* Release 1:

- At least 1 GB of RAM

  To determine the RAM size, enter the following command:

  ```
  # grep MemTotal /proc/meminfo
  ```

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

- The following table describes the relationship between installed RAM and the configured swap space requirement:

  Note: On Linux, hugepages mechanism defines non-swappable memory. If hugepages is used, then you must ensure that you reduce the non swappable portion of the memory from the main memory before calculating swap space.

RAM	Swap Space
Between 1024 MB and 2048 MB	1.5 times the size of RAM
Between 2049 MB and 8192 MB	Equal to the size of RAM
More than 8192 MB	0.75 times the size of RAM

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

  ```
  # grep SwapTotal /proc/meminfo
  ```

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

To determine the available RAM and swap space, enter the following command:

```
# free
```

Note: Oracle recommends that you take multiple values for the available RAM and swap space before finalizing a value. This is because the available RAM and swap space keep changing depending on the user interactions with the computer.

Automatic Memory Management

Starting with Oracle Database 11*g*, the Automatic Memory Management feature requires more shared memory (`/dev/shm`) and file descriptors. The size of the shared memory should be at least the greater of `MEMORY_MAX_TARGET` and `MEMORY_TARGET` for each Oracle instance on the computer. If `MEMORY_MAX_TARGET` or `MEMORY_TARGET` is set to a non zero value, and an incorrect size is assigned to the shared memory, it will result in an `ORA-00845` error at startup. The number of file descriptors for each Oracle instance should be at least `512*`*`PROCESSES`*. Also, the limit of descriptors for each process should be at least 512. If file descriptors are not sized correctly, you will notice `ORA-27123` from various Oracle processes and potentially `Linux Error EMFILE (Too many open files)` errors in non-Oracle processes.

To determine the amount of shared memory available, enter the following command:

```
# df -k /dev/shm/
```

Note: `MEMORY_MAX_TARGET` and `MEMORY_TARGET` cannot be used when `LOCK_SGA` is enabled or with huge pages on Linux.

On the Initialization Parameters page, note that Memory Size (SGA and PGA), which sets the initialization parameter `MEMORY_TARGET` or `MEMORY_MAX_TARGET`. Note that the initialization parameters cannot be greater than the shared memory file system on the operating system. For example, if the shared memory file system allocation on your system is 1 GB, but you set Memory Size (`MEMORY_TARGET`) to 2 GB, then the following error messages are displayed during database startup:

```
ORA-00845: MEMORY_TARGET not supported on this system
ORA-01078: Failure in processing system parameters
```

In addition, if you click All Initialization Parameters and the global database name is longer than 8 characters, then the database name value (in the `DB_NAME` parameter) is truncated to the first eight characters, and the `DB_UNIQUE_NAME` parameter value is set to the global name.

System Architecture

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

```
# uname -m
```

This command displays the processor type. Verify that the processor architecture matches the Oracle software release that you want to install. If you do not see the expected output, then you cannot install the software on this system.

Disk Space Requirements

The following are the disk space requirements for installing Oracle Database 11*g* Release 1:

- Between 150 and 200 MB of disk space in the `/tmp` directory

  To determine the amount of disk space available in the `/tmp` directory, enter the following command:

  ```
  # df -k /tmp
  ```

If there is less than 400 MB of free disk space available in the `/tmp` directory, then complete one of the following steps:

- Delete unnecessary files from the `/tmp` directory to meet the disk space requirement.
- Set the `TMP` and `TMPDIR` environment variables when setting the `oracle` user's environment (described later).
- Extend the file system that contains the `/tmp` directory. If necessary, contact the system administrator for information about extending file systems.

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

```
# df -k
```

- The following table describes the disk space requirements for software files for each installation type on Linux 86:

Installation Type	Requirement for Software Files (GB)
Enterprise Edition	3.47
Standard Edition	3.22
Custom (maximum)	3.45

Between 3.5 GB and 5 GB of disk space for the Oracle software, depending on the installation type

- The following table describes the disk space requirements for each installation type on Linux x86-64:

Installation Type	Requirement for Software Files (GB)
Enterprise Edition	4.35
Standard Edition	3.73
Custom (maximum)	4.54

Between 3.5 GB and 5 GB of disk space for the Oracle software, depending on the installation type

- The following table describes the disk space requirements for each installation type on Linux x86:

Installation Type	Disk Space for Data Files (GB)
Enterprise Edition	1.6
Standard Edition	1.6
Custom (maximum)	1.81

- The following table describes the disk space requirements for each installation type on Linux x86-64:

Installation Type	Disk Space for Data Files (GB)
Enterprise Edition	1.68

Installation Type	Disk Space for Data Files (GB)
Standard Edition	1.48
Custom (maximum)	2.14

Additional disk space, either on a file system or on an Automatic Storage Management disk group is required for the flash recovery area if you choose to configure automated backups.

Recommended Hardware Requirement for SQL Developer

The following table lists the recommended Memory and Display requirements for SQL Developer.

Resource	Recommended
Memory	1 GB RAM (recommended), 256 MB RAM (min)
Display	65536 colors, set to at least 1024 X 768 resolution

Checking the Software Requirements

Depending on the products that you intend to install, verify that the following software are installed on the system.

Note:

- This guide contains information required to install Oracle Database 11*g* Release 1 (11.1) on various platforms of Linux. Ensure that you review information related to the platform on which you intend to install Oracle Database 11*g*.
- Oracle Universal Installer performs checks on the system to verify that it meets the listed requirements. To ensure that these checks pass, verify the requirements before you start Oracle Universal Installer.

Operating System Requirements

The following are the operating system requirements for Oracle Database 11*g* Release 1:

> **Note:** Oracle Universal Installer performs checks to verify that the system meets the listed requirements. To ensure that these checks pass, verify the requirements before you start Oracle Universal Installer.

On Linux x86 and Linux x86-64:

- Asianux 2.0
- Asianux 3.0
- Oracle Enterprise Linux 4.0
- Oracle Enterprise Linux 5.0
- Red Hat Enterprise Linux 4.0
- Red Hat Enterprise Linux 5.0
- SUSE Linux Enterprise Server 10.0

To determine the distribution and version of Linux installed, enter the following command:

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

> **Note:** Only the distributions and versions listed in the earlier list are supported. Do not install the software on other versions of Linux.

Kernel Requirements

The following are the Kernel requirements for Oracle Database 11*g* Release 1:

- On Asianux 2.0, Oracle Enterprise Linux 4.0, and Red Hat Enterprise Linux 4.0:

  2.6.9

- On Asianux 3.0, Oracle Enterprise Linux 5.0, and Red Hat Enterprise Linux 5.0:

  2.6.18

- On SUSE Linux Enterprise Server 10:

  2.6.16.21

To determine whether the required kernel is installed, enter the following command:

```
# uname -r
```

The following is a sample output displayed by running this command on a Red Hat Enterprise Linux 4.0 system:

```
2.6.9-55.0.0.0.2.EL
```

In this example, the output shows the kernel version (`2.6.9`) and errata level (`55.0.0.0.2.EL`) on the system.

If the kernel version does not meet the requirement, then contact the operating system vendor for information about obtaining and installing kernel updates.

Package Requirements

The following are the list of packages required for Oracle Database 11*g* Release 1.

Note: ■

- Oracle recommends that you install the Linux operating system with the default software packages (RPMs). Do not customize the RPMs during installation. This installation includes most required packages, and helps to limit manual checks of package dependencies.
- If you did not perform a default Linux installation, you intend to use LDAP, and you want to use the scripts `odisrvreg`, `oidca`, or `schemasync`, then install the Korn shell RPM for the Linux distribution.

On Linux x86:

Operating System	Requirement

Asianux 2.0, Oracle Enterprise Linux 4.0, and Red Hat Enterprise Linux 4.0	The following packages (or later versions) must be installed:
	`binutils-2.15.92.0.2-18`
	`compat-libstdc++-33.2.3-47.3`
	`elfutils-libelf-0.97-5`
	`elfutils-libelf-devel-0.97-5`
	`gcc-3.4.5-2`
	`gcc-c++-3.4.5-2`
	`glibc-2.3.4-2.19`
	`glibc-common-2.3.4-2.19`
	`glibc-devel-2.3.4-2.19`
	`glibc-headers-2.3.4-2.19`
	`libaio-devel-0.3.105-2`
	`libaio-0.3.105-2`
	`libgcc-3.4.5`
	`libstdc++-3.4.5-2`
	`libstdc++-devel-3.4.5-2`
	`make-3.80-5`
	`sysstat-5.0.5`

Asianux 3.0, Oracle Enterprise Linux 5.0, and Red Hat Enterprise Linux 5.0	The following packages (or later versions) must be installed:
	`binutils-2.17.50.0.6-2.el5`
	`compat-libstdc++-33-3.2.3-61`
	`elfutils-libelf-0.125-3.el5`
	`elfutils-libelf-devel-0.125`
	`gcc-4.1.1-52`
	`gcc-c++-4.1.1-52`
	`glibc-2.5-12`
	`glibc-common-2.5-12`
	`glibc-devel-2.5-12`
	`glibc-headers-2.5-12`
	`libaio-0.3.106`
	`libaio-devel-0.3.106`
	`libgcc-4.1.1-52`
	`libstdc++-4.1.1`
	`libstdc++-devel-4.1.1-52.e15`
	`make-3.81-1.1`
	`sysstat-7.0.0`

Operating System	Requirement
SUSE Linux Enterprise Server 10	The following packages (or later versions) must be installed: binutils-2.16.91.0.5 compat-libstdc++-5.0.7 gcc-4.1.0 glibc-2.4-31.2 glibc-devel-2.4-31.2 ksh-93r-12.9 libaio-0.3.104 libaio-devel-0.3.104 libelf-0.8.5 libgcc-4.1.0 libstdc++-4.1.0 libstdc++-devel-4.1.0 make-3.80 sysstat-6.0.2

On Linux x86-64:

Operating System	Requirement
Asianux 2.0, Oracle Enterprise Linux 4.0, and Red Hat Enterprise Linux 4.0	The following packages (or later versions) must be installed: binutils-2.15.92.0.2 compat-libstdc++-33-3.2.3 compat-libstdc++-33-3.2.3 (32 bit) elfutils-libelf-0.97 elfutils-libelf-devel-0.97 gcc-3.4.5 gcc-c++-3.4.5 glibc-2.3.4-2.19 glibc-2.3.4-2.19 (32 bit) glibc-common-2.3.4 glibc-devel-2.3.4 glibc-devel-2.3.4 (32-bit) libaio-0.3.105 libaio-0.3.105 (32 bit) libaio-devel-0.3.105 libgcc-3.4.5 libgcc-3.4.5 (32-bit) libstdc++-3.4.5 libstdc++-3.4.5 (32 bit) libstdc++-devel 3.4.5 make-3.80 sysstat-5.0.5

Operating System	Requirement
Asianux 3.0, Oracle Enterprise Linux 5.0, and Red Hat Enterprise Linux 5.0	The following packages (or later versions) must be installed: binutils-2.17.50.0.6 compat-libstdc++-33-3.2.3 compat-libstdc++-33-3.2.3 (32 bit) elfutils-libelf-0.125 elfutils-libelf-devel-0.125 gcc-4.1.1 gcc-c++-4.1.1 glibc-2.5-12 glibc-2.5-12 (32 bit) glibc-common-2.5 glibc-devel-2.5 glibc-devel-2.5-12 (32 bit) libaio-0.3.106 libaio-0.3.106 (32 bit) libaio-devel-0.3.106 libgcc-4.1.1 libgcc-4.1.1 (32 bit) libstdc++-4.1.1 libstdc++-4.1.1 (32 bit) libstdc++-devel 4.1.1 make-3.81 sysstat-7.0.0
SUSE Linux Enterprise Server 10	The following packages (or later versions) must be installed: binutils-2.16.91.0.5 compat-libstdc++-5.0.7-22.2 gcc-4.1.0 gcc-c++-4.1.0 glibc-2.4-31.2 glibc-32bit-2.4-31.2 (32 bit) glibc-devel-2.4 glibc-devel-32bit-2.4 (32 bit) libaio-0.3.104 libaio-32bit-0.3.104 (32 bit) libaio-devel-0.3.104 libelf-0.8.5 libgcc-4.1.0 libstdc++-4.1.0 libstdc++-devel-4.1.0 make-3.80 sysstat-6.0.2

To determine whether the required packages are installed, enter commands similar to the following:

```
# rpm -q package_name
```

If a package is not installed, then install it from the Linux distribution media or download the required package version from the Linux vendor's Web site.

Compiler Requirements

Intel C++ Compiler 9.1 or later and the version of GNU C and C++ compilers listed under "Package Requirements" on page 8 are supported with Pro*C/C++, Oracle Call Interface, Oracle C++ Call Interface, and Oracle XML Developer's Kit (XDK) for Oracle Database 11*g* Release 1.

Starting with Oracle Database 11*g* release 1 (11.1.0.7), Pro*COBOL is certified on Red Hat Enterprise Linux 5, SUSE Linux Enterprise Server 10.0, and XEN with Micro Focus Server Express 5.0 WP4.

> **Note:** Intel Compiler v9.1 can be used only with gcc 3.4.5 or gcc 4.0 or gcc 4.1 standard template libraries to build Oracle C++ Call Interface (OCCI) applications.
>
> Oracle XML Developer's Kit is supported with the same compilers as OCCI.

Additional Software Requirements

Depending on the components you want to use, you must ensure that the following software are installed:

- Oracle ODBC Drivers
- Oracle JDBC/OCI Drivers
- Oracle Messaging Gateway
- Browser Requirements
- Oracle XML DB for Oracle Application Express
- PL/SQL Web Toolkit
- Oracle Text

Oracle ODBC Drivers

If you intend to use ODBC, then you should install the most recent ODBC Driver Manager for Linux. You can download and install the Driver Manager from the following link:

`http://www.unixodbc.org`

Linux RPMs are available on the site. You do not require ODBC Driver Manager to install Oracle Database. :

- On Linux x86

  To use ODBC, you must also install the following additional 32-bit ODBC RPMs, depending on your operating system:

  - On Asianux 2, Oracle Enterprise Linux 4, and Red Hat Enterprise Linux 4:

    ```
    unixODBC-2.2.11 (32 bit) or later
    unixODBC-devel-2.2.11 (32 bit) or later
    ```

  - On Asianux 3, Oracle Enterprise Linux 5, and Red Hat Enterprise Linux 5:

    ```
    unixODBC-2.2.11 (32 bit) or later
    unixODBC-devel-2.2.11 (32 bit) or later
    ```

  - On SUSE 10:

    ```
    unixODBC-32bit-2.2.11 (32 bit) or later
    unixODBC-devel-32bit-2.2.11 (32 bit) or later
    ```

- On Linux x86-64

To use ODBC, you must also install the following additional 64-bit ODBC RPMs, depending on your operating system.

– On Asianux 2, Oracle Enterprise Linux 4, and Red Hat Enterprise Linux 4:

```
unixODBC-2.2.11 (32 bit) or later
unixODBC-devel-2.2.11 (64 bit) or later
unixODBC-devel-2.2.11 (64 bit ) or later
```

– On Asianux 3, Oracle Enterprise Linux 5, and Red Hat Enterprise Linux 5:

```
unixODBC-2.2.11 (32 bit) or later
unixODBC-devel-2.2.11 (64 bit) or later
unixODBC-devel-2.2.11 (64 bit ) or later
```

– On SUSE 10:

```
unixODBC-32bit-2.2.11 (32 bit) or later
unixODBC-2.2.11 (64 bit ) or later
unixODBC-devel-2.2.11 (64 bit) or later
```

Oracle JDBC/OCI Drivers

You can use `Sun JDK 1.5.0-06` with the JNDI extension with the Oracle Java Database Connectivity and Oracle Call Interface drivers. However, these are not mandatory for the database installation.

Oracle Messaging Gateway

Oracle Messaging Gateway supports the integration of Oracle Streams Advanced Queuing (AQ) with the following software:

- IBM WebSphere MQ V6.0, client and server, with corrective service diskette 5 (CSD05) or later:

```
MQSeriesClient
MQSeriesServer
MQSeriesRuntime
```

- TIBCO Rendezvous 7.3

If you require a CSD for WebSphere MQ, then refer to the following Web site for download and installation information:

http://www-306.ibm.com/software/integration/wmq/support

Browser Requirements

Web browsers must support Java Script and the HTML 4.0 and CSS 1.0 standards. The following browsers meet these requirements:

- For Oracle Application Express:
  - Microsoft Internet Explorer 6.0 or later version
  - Firefox 1.0 or a later version
- For Oracle Enterprise Manager Database Control:
  - Netscape Navigator 7.2
  - Netscape Navigator 8.1
  - Mozilla version 1.7
  - Microsoft Internet Explorer 6.0 SP2

- Microsoft Internet Explorer 7.0
- Firefox 1.0.4
- Firefox 1.5
- Firefox 2.0

Oracle XML DB for Oracle Application Express

Oracle XML DB must be installed in the Oracle database that you want to use. If you are using a preconfigured database created either during an installation or by Oracle Database Configuration Assistant (DBCA), then Oracle XML DB is already installed and configured.

> **See Also:** *Oracle XML DB Developer's Guide* for more information about manually adding Oracle XML DB to an existing database

PL/SQL Web Toolkit

Oracle Application Express requires the PL/SQL Web Toolkit version 10.1.2.0.6 or later. For instructions on determining the current version of the PL/SQL Web Toolkit, and for instructions on installing version 10.1.2.0.6, review the `README.txt` file contained in the directory `apex/owa`.

Oracle Text

Oracle Text must be installed to use the searchable online Help in Oracle Application Express. By default, Oracle Text is installed as a component Oracle Database.

> **See Also:** *Oracle Text Application Developer's Guide* for more information on Oracle Text

Preinstallation Requirements for Oracle Configuration Manager

During the installation, you are prompted to provide information required to enable Oracle Configuration Manager. When you create a service request with Oracle Support, the configuration information can help to provide a rapid resolution to the service issue.

You can enable Oracle Configuration Manager during or after installation. To enable it during the installation, you must have the following information available:

- Customer Support Identification Number (CSI) that identifies your organization
- Oracle*Metalink* user account name
- Country code associated with the service agreement

Refer to Oracle*Metalink* (`https://metalink.oracle.com`) if there is a registration failures and you are uncertain that the correct country code has been specified. You can find the country associated with the Oracle*MetaLink* account in the Profile section under the Licenses link.

> **See Also:** *Oracle Configuration Manager Installation and Administration Guide* for further information

Checking the Network Setup

Typically, the computer on which you want to install Oracle Database is connected to the network. The computer has local storage, to store the Oracle Database installation.

It also contains a display monitor, and 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 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 checking the hosts file entry in the `nsswitch.conf` file as follows:

   ```
   # cat /etc/nsswitch.conf | grep hosts
   ```

   The output of this command should contain an entry for files.

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

   ```
   # hostname
   ```

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

   myhost`.example.com`

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

   ```
   # domainname
   ```

   This command should not return any results.

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

   ```
   # cat /etc/hosts
   ```

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

   For example:

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

   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 this, ping the host name from the client computers using the short name (host name only) and the full name (host name 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.example.com`, then enter one of the following commands:

In Bourne, Bash, or Korn shell:

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

In C shell:

```
% setenv ORACLE_HOSTNAME somehost.us.example.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 you want to use.

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 you want to install the database 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 should be in the `/etc/hosts` file.

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

If the `ping` command fails, then contact the 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 the 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`.

  Oracle Universal Installer prompts you to specify this group name. If software owner is a member of the group `dba`, then Oracle Universal Installer defaults the OSDBA setting to `dba`. However, you can also choose a different operating system group if required.

- 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.

  In this case, Oracle Universal Installer prompts you to specify the name of this group. The usual name chosen for this group is `oper`.

- The OSASM group (`asmadmin`)

  This feature introduces a new `SYSASM` privilege that is specifically intended for performing Automatic Storage Management administration tasks. Using the `SYSASM` privilege instead of the `SYSDBA` privilege provides a clearer division of responsibility between Automatic Storage Management administration and database administration. OSASM is a new operating system group that is used exclusively for Automatic Storage Management. Members of the OSASM group can connect as `SYSASM` using operating system authentication and have full access to Automatic Storage Management. The usual name chosen for this group is `asmadmin`.

  See Also: "Authentication for Accessing Automatic Storage Management Instances" section in *Oracle Database Storage Administrator's Guide* for more information on `SYSASM` privilege for Automatic Storage Management

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

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

  You must have a group whose members are given access to write to the Oracle Central Inventory (`oraInventory`). The Central Inventory contains the following:

  – A registry of the Oracle home directories (Oracle Database, and Automatic Storage Management) on the system.

  – Installation logs and trace files from installations of Oracle software. These files are also copied to the respective Oracle homes for future reference.

  Other metadata inventory information regarding Oracle installations are stored in the individual Oracle home inventory directories, and are separate from the Central Inventory.

  For new installations, Oracle recommends that you allow OUI to create the Central Inventory directory. By default, if you create an Oracle path in compliance with OFA (Optimal Flexible Architecture) structure, such as `/u01/app`, then the Central Inventory is created in the path `u01/app/oraInventory`, using correct permissions to allow all Oracle installation owners to write to this directory.

- 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 Linux and UNIX* and *Oracle Database Administrator's Guide* for more information on 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 can create the appropriate users and groups in a directory service, for example, Network Information Services (NIS). For information about using directory services, contact the system administrator or refer to the 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 an OSASM Group
- Creating the Oracle Software Owner User

Creating the Oracle Inventory Group

Log in as `root`, and use the following instructions to locate or create the Oracle Inventory group and a software owner:

- Determining Whether the Oracle Inventory Group Exists
- Creating the Oracle Inventory Group

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 (typically, `oinstall`), and the path of the Oracle Inventory directory. An `oraInst.loc` file has contents similar to the following:

```
inventory_loc=central_inventory_location
inst_group=group
```

In the preceding example, *central_inventory_location* is the location of the Oracle Central Inventory, and *group* is the name of the group that has permissions to write to the central inventory.

If you have an existing Oracle Inventory, then ensure that you use the same Oracle Inventory for all Oracle software installations, and ensure that all Oracle software users you intend to use for installation have permissions to write to this directory.

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

```
# more /etc/oraInst.loc
```

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

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

In the previous output example:

- The `inventory_loc` group shows the location of the Oracle Inventory
- The `inst_group` parameter shows the name of the Oracle Inventory group (in this example, `oinstall`).

Creating the Oracle Inventory Group

If the `oraInst.loc` file does not exist, then create the Oracle Inventory group by entering the following command:

```
# /usr/sbin/groupadd oinstall
```

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

To determine whether the OSDBA group exists, enter the following command:

```
# grep OSDBA_group_name /etc/group
```

Note: The default OSDBA group name is dba.

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 already exists.

```
# /usr/sbin/groupadd dba
```

Creating an OSOPER Group (Optional)

Create an OSOPER group only if you want 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 already exists.

```
# /usr/sbin/groupadd oper
```

Creating an OSASM Group

Create an OSASM group only if you want SYSASM as a system privilege that enables the separation of the SYSDBA database administration privilege from the Automatic Storage Management storage administration privilege. If you want to use an OSASM group, then you must create it in the following circumstances:

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

To determine whether the OSASM group exists, enter the following command:

```
# grep OSASM_group_name /etc/group
```

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

```
# /usr/sbin/groupadd asmadmin
```

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, with different group membership, to give database administrative privileges to those groups in a new Oracle Database installation

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 you want to use the existing user or create another `oracle` user. If you want to use the existing user, then ensure that the user's primary group is the Oracle Inventory group and that it is a member of the appropriate `OSDBA` and `OSOPER` groups. Refer to one of the following sections for more information:

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

- If you want to use an existing Oracle software owner user, and the user's primary group is the Oracle Inventory group, then refer to the "Determining Whether an Oracle Software Owner User Exists" section on page 2-20.
- To modify an existing user, refer to the "Modifying an Oracle Software Owner User" section on page 2-21.
- To create a user, refer to "Creating an Oracle Software Owner User" section on page 2-20.

Creating an Oracle Software Owner User

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

1. To create the `oracle` user, enter a command similar to the following:

   ```
   # /usr/sbin/useradd -g oinstall -G dba[,oper] oracle
   ```

   In this command:

   - The `-g` option specifies the primary group, which must be the Oracle Inventory group, for example `oinstall`

- The -G option specifies the secondary groups, which must include the OSDBA group and if required, the OSOPER group (dba or oper)

2. Set the password of the oracle user:

```
# passwd oracle
```

Modifying an Oracle Software Owner User

If the oracle user exists, but its primary group is not *oinstall* or it is not a member of the appropriate OSDBA or OSOPER groups, then enter a command similar to the following to modify it. Specify the primary group using the -g option and any required secondary group using the -G option:

```
# /usr/sbin/usermod -g oinstall -G dba[,oper] oracle
```

Configure Oracle Installation Owner Shell Limits

For information, review "Configuring the oracle User's Environment" on page 2-43

As root, add the following lines to /etc/profile to set shell limits for the user oracle, and for the BASH, Korn, and C shells:

```
umask 022
if [ \$USER = "oracle" ]; then
  if [ \$SHELL = "/bin/ksh" ]; then
  ulimit -p 16384
  ulimit -n 65536
  else
  ulimit -u 16384 -n 65536
```

Configuring Kernel Parameters

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

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

Parameter	Minimum Value	File
semmsl	250	/proc/sys/kernel/sem
semmns	32000	
semopm	100	
semmni	128	
shmall	2097152	/proc/sys/kernel/shmall

Parameter	Minimum Value	File
shmmax	Minimum of the following values: ■ Half the size of the memory ■ 4GB - 1 byte **Note:** The minimum value required for shmmax is 0.5 GB. However, Oracle recommends that you set the value of shmmax to 2.0 GB for optimum performance of the system.	/proc/sys/kernel/shmmax
shmmni	4096	/proc/sys/kernel/shmmni
file-max	512 * *PROCESSES*	/proc/sys/fs/file-max
ip_local_port_range	Minimum:1024 Maximum: 65000	/proc/sys/net/ipv4/ip_local_port_range
rmem_default	4194304	/proc/sys/net/core/rmem_default
rmem_max	4194304	/proc/sys/net/core/rmem_max
wmem_default	262144	/proc/sys/net/core/wmem_default
wmem_max	262144	/proc/sys/net/core/wmem_max
tcp_wmem	262144	/proc/sys/net/ipv4/tcp_wmem
tcp_rmem	4194304	/proc/sys/net/ipv4/tcp_rmem

To display the current value specified for these kernel parameters, and to change them if necessary, use the following steps:

- Enter the commands shown in the following table to display the current values of the kernel parameters, make a note of these values and identify any values that you must change:

Parameter	Command
semmsl, semmns, semopm, and semmni	# /sbin/sysctl -a | grep sem This command displays the value of the semaphore parameters in the order listed.
shmall, shmmax, and shmmni	# /sbin/sysctl -a | grep shm This command displays the details of the shared memory segment sizes.
file-max	# /sbin/sysctl -a | grep file-max This command displays the maximum number of file handles.
ip_local_port_range	# /sbin/sysctl -a | grep ip_local_port_range This command displays a range of port numbers.
rmem_default	# /sbin/sysctl -a | grep rmem_default
rmem_max	# /sbin/sysctl -a | grep rmem_max
wmem_default	# /sbin/sysctl -a | grep wmem_default

Parameter	Command
wmem_max	# /sbin/sysctl -a | grep wmem_max
tcp_wmem	# /sbin/sysctl -a | grep tcp_wmem
tcp_rmem	# /sbin/sysctl -a | grep tcp_rmem

- If the value of any kernel parameter is different from the minimum value, then complete the following procedure:

  1. Using any text editor, 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 that you want to change. For the semaphore parameters (`kernel.sem`), you must specify all four values. However, if any of the current values are larger than the minimum value, then specify the larger value.

     ```
     fs.file-max = 512 * PROCESSES
     kernel.shmall = 2097152
     kernel.shmmax = 2147483648
     kernel.shmmni = 4096
     kernel.sem = 250 32000 100 128
     net.ipv4.ip_local_port_range = 1024 65000
     net.core.rmem_default = 4194304
     net.core.rmem_max = 4194304
     net.core.wmem_default = 262144
     net.core.wmem_max = 262144
     net.ipv4.tcp_wmem = 262144 262144 262144
     net.ipv4.tcp_rmem = 4194304 4194304 4194304
     ```

     Note: The minimum value required for `shmmax` is 0.5 GB. However, Oracle recommends that you set the value of `shmmax` to 2.0 GB for optimum performance of the system.

     By specifying the values in the `/etc/sysctl.conf` file, they persist when you restart the system. However, on SUSE Linux Enterprise Server systems, enter the following command to ensure that the system reads the `/etc/sysctl.conf` file when it restarts:

     ```
     # /sbin/chkconfig boot.sysctl on
     ```

  2. Enter the following command to change the current values of the kernel parameters:

     ```
     # /sbin/sysctl -p
     ```

     Review the output from this command to verify that the values are correct. If the values are incorrect, edit the `/etc/sysctl.conf` file, then enter this command again.

  3. Enter the command `/sbin/sysctl -a` to confirm that the values are set correctly.

4. On SUSE systems only, enter the following command to cause the system to read the `/etc/sysctl.conf` file when it restarts:

```
# /sbin/chkconfig boot.sysctl on
```

5. On SUSE systems only, you must enter the GID of the *`oinstall`* group as the value for the parameter `/proc/sys/vm/hugetlb_shm_group`. Doing this grants members of *`oinstall`* a group permission to create shared memory segments.

   For example, where the *`oinstall`* group GID is 501:

```
# echo 501 > /proc/sys/vm/hugetlb_shm_group
```

   After running this command, use `vi` to add the following text to `/etc/sysctl.conf`, and enable the `boot.sysctl` script to run on system restart:

```
vm.hugetlb_shm_group=501
```

   Note: Only one group can be defined as the `vm.hugetlb_shm_group`.

6. After updating the values of kernel parameters in the `/etc/sysctl.conf` file, either restart the computer, or run the command `sysctl -p` to make the changes in the `/etc/sysctl.conf` file available in the active kernel memory.

Setting Shell Limits for the `oracle` User

To improve the performance of the software, you must increase the following shell limits for the `oracle` user:

Shell Limit	Item in limits.conf	Hard Limit
Maximum number of open file descriptors	`nofile`	65536
Maximum number of processes available to a single user	`nproc`	16384

To increase the shell limits:

1. Add the following lines to the `/etc/security/limits.conf` file:

```
oracle              soft    nproc   2047
oracle              hard    nproc   16384
oracle              soft    nofile  1024
oracle              hard    nofile  65536
```

2. Add or edit the following line in the `/etc/pam.d/login` file, if it does not already exist:

```
session    required     pam_limits.so
```

3. Depending on the `oracle` user's default shell, make the following changes to the default shell start-up file:

- For the Bourne, Bash, or Korn shell, add the following lines to the `/etc/profile` file (or the file on SUSE Linux Enterprise Server systems `/etc/profile.local`):

```
if [ $USER = "oracle" ]; then
        if [ $SHELL = "/bin/ksh" ]; then
              ulimit -p 16384
              ulimit -n 65536
        else
              ulimit -u 16384 -n 65536
        fi
fi
```

- For the C shell (`csh` or `tcsh`), add the following lines to the `/etc/csh.login` file (or the file on SUSE Linux Enterprise Server systems `/etc/csh.login.local`):

```
if ( $USER == "oracle" ) then
        limit maxproc 16384
        limit descriptors 65536
endif
```

Refer to the "Identifying Required Software Directories" section to continue.

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. On Linux 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 can 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`.

Note: If you start a database instance using `spfile` with `ORACLE_BASE` environment variable set, then its value is automatically stored in `spfile`. If you unset `ORACLE_BASE` environment variable subsequently and start the instance afresh, then database uses the value of Oracle base stored in `spfile`.

You need to specify the Oracle base folder that contains all Oracle products.

Note: If you have an existing Oracle base, then you can select it from the Use existing list. By default, the list contains the existing value for Oracle base preselected. Refer to "Installing the Oracle Database Software" on page 2-6 for further information.

If you do not have an Oracle base, then you can create one by editing the text in the list box.

You can use the same Oracle base directory for more than one installation 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 are the example of Oracle base directories that can exist on the same system:

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

Oracle Inventory Directory

The Oracle Inventory directory (`oraInventory`) stores an inventory of all software installed on the system. It is required and shared by all Oracle software installations on a single system. If you have an existing Oracle Inventory path, then Oracle Universal Installer continues to use that Oracle Inventory.

The first time you install Oracle software on a system, Oracle Universal Installer checks if you have created an OFA-compliant directory structure with the format `u[01-09]/app`, such as `/u01/app`, and that the user running the installation has permissions to write to that path. If this is true, then Oracle Universal Installer creates the Oracle Inventory directory similar to `/u[01-09]/app/oraInventory`. For example:

```
/u01/app/oraInventory
```

If you have set the environment variable `ORACLE_BASE` for the `oracle` user, then Oracle Universal Installer creates the Oracle Inventory directory similar to `$ORACLE_BASE/../oraInventory`. For example, if `ORACLE_BASE` is set to `/opt/oracle/11`, then the Oracle Inventory directory is created similar to `/opt/oracle/oraInventory`.

If you have neither created an OFA-compliant path nor set `ORACLE_BASE`, then the Oracle Inventory directory is placed in the home directory of the user that is performing the installation. For example:

```
/home/oracle/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 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/11.1.0/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.

Note: 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 that comply with OFA (Optimal Flexible Architecture) 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. Refer to Creating the Oracle Inventory Group on page 2-18 for more information.

> **Note:** Oracle recommends that you do not put the `oraInventory` directory under Oracle base for a new installation. However, if you have an existing installation, then you should follow the steps suggested in this section.

- Identifying an existing Oracle home directory

  Enter the following command to display 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/11.1.0/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 that you want to use are valid choices for an Oracle base directory. If you intend to use the `oracle` user to install the software, then you can choose one of the following directories listed in the previous example:

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

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

- Identifying an existing Oracle base directory

  After you have located the Oracle home directory, you can run the following command to confirm the location of Oracle base:

  ```
  cat inventory/ContentsXML/oraclehomeproperties.xml
  ```

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

- It should 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 will contain only software files.	Up to 3 GB
The Oracle base directory will contain both software and database files (not recommended for production databases).	Up to 5.4 GB

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

```
# df -k 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 2-29.

  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.

To identify an appropriate file system:

1. To determine the free disk space on each mounted file system use the following command:

   ```
   # df -k
   ```

2. From the display, identify a file system that has appropriate free space.

   The file system that you identify can be a local file system, a cluster file system, or an NFS file system on a certified NAS device.

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

   For example:

   ```
   # mkdir -p /u01/app
   # chown -R oracle:oinstall /u01/app
   # chmod -R 775 /u01/app/
   ```

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

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 that you want to use for Oracle Database files. If you want to enable automated backups during the installation, then you must also choose the storage option that you want to use for recovery files (the flash recovery area). You do not have to use the same storage option for each file type.

Note: Database files and recovery files are supported on file systems and Automatic Storage Management.

Use the following guidelines when choosing the storage options that you want to use for each file type:

- You can choose any combination of the supported storage options for each file type.
- Oracle recommends that you choose Automatic Storage Management as the storage option for database and recovery files.
- For more information about these storage options, refer to the "Database Storage Options" section on page 1-12.

For information about how to configure disk storage before you start the installation, refer to one of the following sections depending on your choice:

- 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 2-30.
- To use Automatic Storage Management for database or recovery file storage, refer to the "Preparing Disk Groups for an Automatic Storage Management Installation" section on page 2-32.

Creating Directories for Oracle Database or Recovery Files

This section contains the following topics:

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 more than one file system to store the database files:
  - If you want to use a single file system, then 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 more than one physical device and implement the stripe-and-mirror-everything (SAME) methodology.
  - If you want to use more than one file system, then choose file systems on separate physical devices that are dedicated to the database.

    This method enables you to distribute physical input-output operations and create separate control files on different devices for increased reliability. It also enables you to fully implement the OFA guidelines described in Appendix D, "Optimal Flexible Architecture". You must choose either the Advanced database creation option or the Custom installation type during the installation to implement this method.

- If you intend to create a preconfigured database during the installation, then the file system (or file systems) that you choose must have at least 1.5 GB of free disk space.

  For production databases, you must estimate the disk space requirement depending on the use that you want to make of the database.
- For optimum performance, the file systems that you choose should 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.

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 following to determine the free disk space on each mounted file system:

   ```
   # df -k
   ```

2. From the display, identify the file systems that you want to use:

File Type	File System Requirements
Database files	Choose either: ■ A single file system with at least 1.5 GB of free disk space ■ Two or more file systems with at least 1.5 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 more than one type of file, 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
     ```

     The default location for Database file directory is `$ORACLE_BASE/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
```

The default flash recovery area is `$ORACLE_BASE/flash_recovery_area`. However, Oracle recommends that you keep the flash recovery area on a separate physical disk than that of the database file directory. This will enable you use the flash recovery area to retrieve data if the disk containing `oradata` is unusable due to any reasons.

5. If you also want to use Automatic Storage Management for storage, then refer to the following section:

   "Preparing Disk Groups for an Automatic Storage Management Installation" on page 2-32

   Otherwise, refer to the "Stopping Existing Oracle Processes" section on page 2-42.

Preparing Disk Groups for an Automatic Storage Management Installation

This section describes how to configure disks for use with Automatic Storage Management. Before you configure the disks, you must determine the number of disks and the amount of free disk space that you require. The following sections describe how to identify the requirements and configure the disks on each platform:

- General Steps for Configuring Automatic Storage Management
- Step 1: Identifying Storage Requirements for Automatic Storage Management
- Step 2: Using an Existing Automatic Storage Management Disk Group
- Step 3: Creating DAS or SAN Disk Partitions for Automatic Storage Management
- Step 4: Configuring Disks for Automatic Storage Management

General Steps for Configuring Automatic Storage Management

The following are the general steps to configure Automatic Storage Management:

1. Identify the storage requirements of the site.
2. Optionally, use an existing Automatic Storage Management disk group.
3. If you are creating an Automatic Storage Management disk group, create partitions for DAS or SAN disks.
4. Use one of the following methods to complete the Automatic Storage Management configuration:
   - If you plan to install Oracle Database using interactive mode, Oracle Universal Installer prompts you for the Automatic Storage Management disk configuration information during the installation.
   - If you plan to install Oracle Database using noninteractive mode, you should manually configure the disks before performing the installation.

Step 1: Identifying Storage Requirements for Automatic Storage Management

To identify the storage requirements for using Automatic Storage Management, you must determine the number of devices and the amount of free disk space that you require. To complete this task:

1. Determine whether you want to use Automatic Storage Management for Oracle Database files, recovery files, or both.

Note: You do not have to use the same storage mechanism for data files and recovery files. You can use the file system for one file type and Automatic Storage Management for the other. If you plan to use Automatic Storage Management for both data files and recovery files, then you should create separate Automatic Storage Management disk groups for the data files and the recovery files.

If you plan to enable automated backups during the installation, then you can choose Automatic Storage Management as the storage mechanism for recovery files by specifying an Automatic Storage Management disk group for the flash recovery area. Depending on how you choose to create a database during the installation, you have the following options:

- If you select an installation method that runs Oracle Database Configuration Assistant in an interactive mode, by choosing the Advanced database configuration option for example, then you can decide whether you want to use the same Automatic Storage Management disk group for database files and recovery files, or you can choose to use different disk groups for each file type. Ideally, you should create separate Automatic Storage Management disk groups for data files and for recovery files.

  The same choice is available to you if you use Oracle Database Configuration Assistant after the installation to create a database.

- If you select an installation type that runs Oracle Database Configuration Assistant in noninteractive mode, then you must use the same Automatic Storage Management disk group for data files and recovery files.

2. Choose the Automatic Storage Management redundancy level that you want to use for each Automatic Storage Management disk group that you create.

   The redundancy level that you choose for the Automatic Storage Management disk group determines how Automatic Storage Management mirrors files in the disk group and determines the number of disks and amount of disk space that you require, as follows:

   - External redundancy

     An external redundancy disk group requires a minimum of one disk device. The effective disk space in an external redundancy disk group is the sum of the disk space in all of its devices.

     This option does not allow Automatic Storage Management to mirror the contents of the disk group. Oracle recommends that you select this redundancy level either when the disk group contains devices, such as RAID devices, that provide their own data protection or when the database does not require an uninterrupted access to data.

   - Normal redundancy

     In a normal redundancy disk group, to increase performance and reliability, Automatic Storage Management by default uses two-way mirroring. A normal redundancy disk group requires a minimum of two disk devices (or two failure groups). The effective disk space in a normal redundancy disk group is half the sum of the disk space in all of its devices.

     For most installations, Oracle recommends that you use normal redundancy disk groups.

   - High redundancy

The contents of the disk group are three-way mirrored by default. To create a disk group with high redundancy, you must specify at least three failure groups (a minimum of 3 devices).

Although high-redundancy disk groups provide a high level of data protection, you must consider the higher cost of additional storage devices before deciding to use this redundancy level.

3. Determine the total amount of disk space that you require for the database files and recovery files.

   If an Automatic Storage Management instance is already running on the system, then you can use an existing disk group to meet these storage requirements. If necessary, you can add disks to an existing disk group during the installation.

   Use the following table to determine the minimum number of disks and the minimum disk space requirements for the installation:

Redundancy Level	Minimum Number of Disks	Data Files	Recovery Files	Both File Types
External	1	1.15 GB	2.3 GB	3.45 GB
Normal	2	2.3 GB	4.6 GB	6.9 GB
High	3	3.45 GB	6.9 GB	10.35 GB

   The following step describes how to identify existing disk groups and determine the free disk space that they contain.

4. Identify failure groups for the Automatic Storage Management disk group devices.

   Note: You need to perform this step only when you intend to use an installation method that runs Oracle Database Configuration Assistant in an interactive mode. For example, if you intend to choose the Custom installation type or the Advanced database configuration option. Other installation types do not enable you to specify failure groups.

   If you intend to use a normal or high redundancy disk group, then you can further protect the database against hardware failure by associating a set of disk devices in a custom failure group. By default, each device comprises its failure group. However, if two disk devices in a normal redundancy disk group are attached to the same SCSI controller, then the disk group becomes unavailable if the controller fails. The controller in this example is a single point of failure.

   To avoid failures of this type, you can use two SCSI controllers, each with two disks, and define a failure group for the disks attached to each controller. This configuration would enable the disk group to tolerate the failure of one SCSI controller.

   Note: If you define custom failure groups, then you must specify a minimum of two failure groups for normal redundancy disk groups and three failure groups for high redundancy disk groups.

5. If you are sure that a suitable disk group does not exist on the system, then install or identify appropriate disk devices to add to a new disk group. Apply the following guidelines when identifying appropriate disk devices:
   - All the devices in an Automatic Storage Management disk group should be the same size and have the same performance characteristics.
   - Do not specify more than one partition on a single physical disk as a disk group device. Automatic Storage Management expects each disk group device to be on a separate physical disk.
   - Oracle does not recommend the use of logical volume as a device in an Automatic Storage manger because the logical volume is capable of hiding the physical disk architecture which prevents Automatic Storage Manager from optimizing I/O across physical devices.

   See Also: "Step 4: Configuring Disks for Automatic Storage Management" on page 2-37 for information about completing this task

Step 2: Using an Existing Automatic Storage Management Disk Group

Note: This is an optional step.

If you want to store either database or recovery files in an existing Automatic Storage Management disk group, then you have the following choices, depending on the installation method that you select:

- To run Oracle Database Configuration Assistant in an interactive mode (for example, choosing the Advanced database configuration option), you can decide to create a disk group or use an existing one.

  The same choice is available to you if you use Oracle Database Configuration Assistant after the installation to create a database.
- If you select an installation method that runs Oracle Database Configuration Assistant in noninteractive mode, then you must choose an existing disk group for the new database; you cannot create a disk group. However, you can add disk devices to an existing disk group if it has insufficient free space for your requirements.

Note: The Automatic Storage Management instance that manages the existing disk group can be running in a different Oracle home directory.

To determine whether an existing Automatic Storage Management disk group exists, or to determine whether there is sufficient disk space in a disk group, you can use Oracle Enterprise Manager Grid Control or Database Control. Alternatively, you can use the following procedure:

1. View the contents of the `oratab` file to determine whether an Automatic Storage Management instance is configured on the system:

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

   If an Automatic Storage Management instance is configured on the system, then the `oratab` file should contain a line similar to the following:

```
+ASM:oracle_home_path:N
```

In this example, `+ASM` is the system identifier (`SID`) of the Automatic Storage Management instance and *`oracle_home_path`* is the Oracle home directory where it is installed. By convention, the `SID` for an Automatic Storage Management instance begins with a plus sign.

2. Open a shell prompt and temporarily set the `ORACLE_SID` and `ORACLE_HOME` environment variables to specify the appropriate values for the Automatic Storage Management instance that you want to use.

   For example, if the Automatic Storage Management `SID` is named `OraDB11g+ASM` and is located in the `asm` subdirectory of the `ORACLE_BASE` directory, then enter the following commands to create the required settings:

   - Bourne, Bash, or Korn shell:

     ```
     $ ORACLE_SID=OraDB11g+ASM
     $ export ORACLE_SID
     $ ORACLE_HOME=/u01/app/oracle/product/11.1.0/asm
     $ export ORACLE_HOME
     ```

   - C shell:

     ```
     % setenv ORACLE_SID OraDB11g+ASM
     % setenv ORACLE_HOME /u01/app/oracle/product/11.1.0/asm
     ```

3. By using SQL*Plus, connect to the Automatic Storage Management instance as the `SYS` user with `SYSASM` privilege and start the instance, if necessary:

   ```
   # $ORACLE_HOME/bin/sqlplus
   SQL> CONNECT SYS as SYSASM
   Enter password: SYS_password
   SQL> STARTUP
   ```

4. Enter the following command to display the existing disk groups, their redundancy level, and the amount of free disk space in each one:

   ```
   SQL> SELECT NAME,TYPE,TOTAL_MB,FREE_MB FROM V$ASM_DISKGROUP;
   ```

5. From the output, identify a disk group with the appropriate redundancy level and note the free space that it contains.
6. If necessary, install or identify the additional disk devices required to meet the storage requirements listed in the previous section.

Note: If you are adding devices to an existing disk group, then Oracle recommends that you use devices that have the same size and performance characteristics as the existing devices in that disk group.

Step 3: Creating DAS or SAN Disk Partitions for Automatic Storage Management

In order to use a DAS or SAN disk in Automatic Storage Management, the disk must have a partition table. Oracle recommends creating exactly one partition for each disk containing the entire disk.

Note: You can use any physical disk for Automatic Storage Management, as long as it is partitioned.

Step 4: Configuring Disks for Automatic Storage Management

Oracle provides an Automatic Storage Management library driver that you can use to simplify the configuration and management of the disk devices that you want to use with Automatic Storage Management. A disk that is configured for Automatic Storage Management is known as a candidate disk.

If you intend to use Automatic Storage Management for database storage, then Oracle recommends that you install the Automatic Storage Management library driver (ASMLIB) and associated utilities and use them to configure the devices that you want to include in an Automatic Storage Management disk group.

> **Note:** If you choose to configure disks using the Automatic Storage Management library driver, then you must change the default disk discovery string to `ORCL:*`. These disks would be discovered if the diskstring is either set to `ORCL:*` or is left empty ("").

Configuring Disks for Automatic Storage Management Using the Automatic Storage Management Library Driver (ASMLIB)

To use the Automatic Storage Management library driver to configure Automatic Storage Management devices, complete the following tasks:

- Installing and Configuring the Automatic Storage Management Library Driver Software
- Configuring the Disk Devices to Use the Automatic Storage Management Library Driver
- Administering the Automatic Storage Management Library Driver and Disks

Installing and Configuring the Automatic Storage Management Library Driver Software

To install and configure the Automatic Storage Management library driver software:

1. Enter the following command to determine the kernel version and architecture of the system:

   ```
   # uname -rm
   ```

2. If necessary, download the required Automatic Storage Management library driver packages from the Oracle Technology Network Web site:

   `http://www.oracle.com/technology/tech/linux/asmlib/index.html`

   > **Note:** Automatic Storage Management library driver packages for some kernel versions are available on the Oracle Database installation media in the `database/RPMS/asmlib` directory. However, Oracle recommends that you check the Oracle Technology Network Web site for the most up-to-date packages.

   You must install the following packages, where *`version`* is the version of the Automatic Storage Management library driver, *`arch`* is the system architecture, and *`kernel`* is the version of the kernel that you are using:

   ```
   oracleasm-support-version.arch.rpm
   oracleasm-kernel-version.arch.rpm
   oracleasmlib-version.arch.rpm
   ```

3. Enter a command similar to the following to install the packages:

```
# sudo rpm -Uvh oracleasm-support-version.arch.rpm \
           oracleasm-kernel-version.arch.rpm \
           oracleasmlib-version.arch.rpm
```

   For example, if you are using the Red Hat Enterprise Linux AS 5.0 enterprise kernel on an x86 system, then enter a command similar to the following:

```
# sudo rpm -Uvh oracleasm-support-1.0.0-1.i386.rpm \
           oracleasm-2.6.9-e-enterprise-1.0.0-1.i686.rpm \
           oracleasmlib-1.0.0-1.i386.rpm
```

4. Enter a command similar to the following to determine the UID of the Oracle software owner user that you are using for this installation (typically `oracle`) and the GID of the OSASM group (typically asm):

```
# id oracle
```

5. Enter the following command to run the `oracleasm` initialization script with the `configure` option:

```
# /etc/init.d/oracleasm configure
```

6. Enter the following information in response to the prompts that the script displays:

Prompt	Suggested Response
Default UID to own the driver interface:	Specify the UID of the Oracle software owner user (`oracle`).
Default GID to own the driver interface:	Specify the GID of the OSASM group (`asm`).
Start Oracle Automatic Storage Management Library driver on start (y/n):	Enter `y` to start the Oracle Automatic Storage Management library driver when the system starts.

Configuring the Disk Devices to Use the Automatic Storage Management Library Driver

To configure the disk devices that you want to use in an Automatic Storage Management disk group:

1. If you intend to use IDE, SCSI, or RAID devices in the Automatic Storage Management disk group, then:
   a. If necessary, install or configure the disk devices that you intend to use for the disk group and restart the system.
   b. To identify the device name for the disks that you want to use, enter the following command:

```
# /sbin/fdisk -l
```

      Depending on the type of disk, the device name can vary.

Disk Type	Device Name Format	Description
IDE disk	/dev/hd*xn*	In this example, *x* is a letter that identifies the IDE disk and *n* is the partition number. For example, /dev/hda is the first disk on the first IDE bus.
SCSI disk	/dev/sd*xn*	In this example, *x* is a letter that identifies the SCSI disk and *n* is the partition number. For example, /dev/sda is the first disk on the first SCSI bus.
RAID disk	/dev/rd/c*x*d*y*p*z* /dev/ida/c*x*d*y*p*z*	Depending on the RAID controller, RAID devices can have different device names. In the examples shown, *x* is a number that identifies the controller, *y* is a number that identifies the disk, and *z* is a number that identifies the partition. For example, /dev/ida/c0d1 is the second logical drive on the first controller.

To include devices in a disk group, you can specify either whole-drive device names or partition device names.

Note: Oracle recommends that you create a single whole-disk partition on each disk that you want to use.

c. Use either `fdisk` or `parted` to create a single whole-disk partition on the disk devices that you want to use.

2. Enter a command similar to the following to mark a disk as an Automatic Storage Management disk:

```
# /etc/init.d/oracleasm createdisk DISK1 /dev/sdb1
```

In this example, `DISK1` is a name that you want to assign to the disk.

Note:

- If you are using a multi-pathing disk driver with Automatic Storage Management, then ensure that you specify the correct logical device name for the disk.

  The disk names that you specify can contain uppercase letters, numbers, and the underscore character. They must start with an uppercase letter.
- To create a database during the installation using the Automatic Storage Management library driver, you must change the default disk discovery string to `ORCL:*`. These disks would be discovered if the diskstring is either set to `ORCL:*` or is left empty.

Administering the Automatic Storage Management Library Driver and Disks

To administer the Automatic Storage Management library driver and disks, use the `oracleasm` initialization script with the following options:

Option	Description
`configure`	Use the `configure` option to reconfigure the Automatic Storage Management library driver, if necessary: `# /etc/init.d/oracleasm configure`
`enable` `disable`	Use the `disable` and `enable` options to change the behavior of the Automatic Storage Management library driver when the system starts. The `enable` option causes the Automatic Storage Management library driver to load when the system starts: `# /etc/init.d/oracleasm enable`

`start`
`stop`
`restart`
	Use the `start`, `stop`, and `restart` options to load or unload the Automatic Storage Management library driver without restarting the system:
	`# /etc/init.d/oracleasm restart`

`createdisk`	Use the `createdisk` option to mark a disk device for use with the Automatic Storage Management library driver and give it a name: `# /etc/init.d/oracleasm createdisk` *`DISKNAME devicename`*
`deletedisk`	Use the `deletedisk` option to unmark a named disk device: `# /etc/init.d/oracleasm deletedisk` *`DISKNAME`* **Note:** Do not use this command to unmark disks that are being used by an Automatic Storage Management disk group. You must drop the disk from the Automatic Storage Management disk group before you unmark it.
`querydisk`	Use the `querydisk` option to determine whether a disk device or disk name is being used by the Automatic Storage Management library driver: `# /etc/init.d/oracleasm querydisk {`*`DISKNAME`* `|` *`devicename`*`}`
`listdisks`	Use the `listdisks` option to list the disk names of marked Automatic Storage Management library driver disks: `# /etc/init.d/oracleasm listdisks`
`scandisks`	Use the `scandisks` option to enable cluster nodes to identify which shared disks have been marked as Automatic Storage Management library driver disks on another node: `# /etc/init.d/oracleasm scandisks`

Configuring Disk Devices for Oracle Database

The `O_DIRECT` parameter enables direct read and writes to block devices, avoiding kernel overhead. With Oracle Database Release 10.2 and later, Oracle Database files are configured by default to use direct input/output.

With the 2. 6 kernel or later for Red Hat Enterprise Linux, Oracle Enterprise Linux, and SUSE Enterprise Server, you must create a permissions file to maintain permissions on Oracle database files. If you do not create this permissions file, then permissions on disk devices revert to their default values, `root:disk`, and Oracle database fails to start. Use the following steps to set the permissions file number:

- On Asianux 2, Red Hat Enterprise Linux 4, and Oracle Enterprise Linux 4, you must create a permissions file number that is lower than 50.
- On Asianux 3, Red Hat Enterprise Linux 5, Oracle Enterprise Linux 5, or SUSE Enterprise Linux 10, you must create a permissions file number that is higher at 50.

To configure a permissions file for disk devices, complete the following tasks:

- Example of Creating a `Udev` Permissions File for Oracle Database
- Example of Configuring Block Device Storage for Oracle Database

See Also: *Oracle Clusterware Installation Guide for Linux* for information about configuring storage for Oracle database files on shared storage devices.

Example of Creating a `Udev` Permissions File for Oracle Database

The procedure to create a permissions file to grant *`oinstall`* group members write privileges to block devices is as follows:

1. Log in as `root`.
2. Change to the `/etc/udev/permissions.d` directory:

   ```
   # cd /etc/udev/permissions.d
   ```

3. Start a text editor, such as `vi`, and enter the partition information where you want to place the data files and voting disk files, using the syntax *device*[*partitions*]:root:oinstall:0640. Oracle recommends that you place the data files on separate physical disks. For example, to grant *`oinstall`* members access to SCSI disks to place data files on `sda` and `sdb`, and to grant the Oracle Database owner permissions to place voting disks on sdF, sdG and sdE, add the following information to the file:

   Binary Files

   ```
   sdb:oracle:oinstall:0640
   sdc:oracle:oinstall:0640
   sda:oracle:oinstall:0640
   ```

   Data Files/Flash Recovery Files

   ```
   sdE:oracle:oinstall:0640
   sdF:oracle:oinstall:0640
   sdG:oracle:oinstall:0640
   ```

4. Save the file as `49-oracle.permissions` on a Red Hat and Oracle Enterprise Linux 4 system and `51-oracle.permissions` on a SUSE Linux Enterprise Server 10 system
5. Using the following command, assign the permissions in the `udev` file to the devices:

   ```
   # /sbin/udevstart
   ```

Example of Configuring Block Device Storage for Oracle Database

The following is the procedure to create partitions for Oracle Database files on block devices:

1. Log in as `root`

2. Enter the `fdisk` command to format a specific storage disk. For example, `/sbin/fdisk /dev/sdb`
3. Create a partition, and make the partition 280 MB in size for both data files and voting disk partitions.
4. Use the command similar to the following to update the kernel partition table for the shared storage device:

   ```
   /sbin/partprobe diskpath
   ```

The following is an example of how to use `fdisk` to create one partition on a shared storage block disk device for a data file:

```
$ sudo sh
Password:
# /sbin/fdisk /dev/sdb
The number of cylinders for this disk is set to 1024.
Command (m for help): n
Command action
  e   extended
  P   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-1024, default 1):
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-4462, default 1)
Using default value 1
Last cylinder or +size or +sizeM or +sizeK (1-1024, default 4462): using default
value 4462

Command (m for help):w

The partition table has been altered!
Calling ioctl () to re-read partition table.
Synching disks.
# exit
$ ssh remotenode
Last login Wed Feb 21 20:23:01 from localnode
$ sudo sh
Password:
# /sbin/partprobe /dev/sdb1
```

Stopping Existing Oracle Processes

> **Note:** If you are installing additional Oracle Database 11*g* 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, 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 listeners 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 tnslsnr
   ```

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

   ```
   ... oracle_home1/bin/tnslsnr 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 run, then refer to the "Configuring the oracle User's Environment" section on page 2-43 to continue.

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

   - Bourne, Bash, or Korn shell:

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

   - 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, for example, an X terminal (`xterm`).
2. Enter the following command to ensure that X Window applications can display on this system:

   ```
   $ xhost fully_qualified_remote_host_name
   ```

   For example:

   ```
   $ xhost somehost.us.example.com
   ```

3. 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.
4. If you are not logged in as the `oracle` user, then switch user to `oracle`:

   ```
   $ su - oracle
   ```

5. To determine the default shell for the `oracle` user, enter the following command:

   ```
   $ echo $SHELL
   ```

6. To run the shell startup script, enter one of the following commands:
   - Bash shell:

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

   - Bourne or Korn shell:

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

   - C shell:

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

7. If you are not installing the software on the local computer, then run the following command on the remote computer to set the `DISPLAY` variable:
   - Bourne, Bash or Korn shell:

     ```
     $ export DISPLAY=local_host:0.0
     ```

   - C shell:

     ```
     % setenv DISPLAY local_host:0.0
     ```

   In this example, `local_host` is the host name or IP address of the local computer that you want to use to display Oracle Universal Installer.

   Run the following command on the remote computer to check if the shell and the DISPLAY environmental variable are set correctly:

   ```
   echo $SHELL
   echo $DISPLAY
   ```

   Now to enable X applications, run the following commands on the local computer:

   ```
   $ xhost + fully_qualified_remote_host_name
   ```

   To verify that X applications display is set properly, run a X11 based program that comes with the operating system such as `xclock`:

   ```
   $ xclock_path
   ```

In this example, xclock_path is the directory path. For example, you can find xclock at /usr/X11R6/bin/xclocks. If the DISPLAY variable is set properly, then you can see xclock on your computer screen.

See Also: PC-X Server or Operating System vendor documents for further assistance.

8. 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. To determine the free disk space on each mounted file system use the following command:

   ```
   # df -k /tmp
   ```

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

   ```
   $ sudo mkdir /mount_point/tmp
   $ sudo chmod a+wr /mount_point/tmp
   # exit
   ```

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

      * Bourne, Bash, or Korn shell:

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

      * C shell:

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

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

   - Bourne, Bash, or Korn shell:

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

   - 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 that you want to call the database (typically no more than five characters).

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

   - Bourne, Bash, or Korn shell:

   ```
   $ unset ORACLE_HOME
   ```

```
$ unset TNS_ADMIN
```

- 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.

3

Installing Oracle Database

The Oracle Database software is available on installation media 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

Preinstallation Considerations

After reviewing the information in Chapter 1, "Overview of Oracle Database Installation" and completing the tasks listed in Chapter 2, "Oracle Database Preinstallation Requirements", consider the following case:

Performing Multiple Oracle Database Installations in Noninteractive Mode

If you need to perform multiple installations of Oracle Database, you may want to 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 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.

- Reinstalling Oracle Software

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

- Products requiring a custom installation

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

  - Oracle Database Vault
  - Oracle Connection Manager
  - Oracle Label Security

  Note: Before you perform a custom installation, make sure all databases in the Oracle home where you want to install Oracle Label Security are shut down.

  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.

- Installations on a Cluster

  If Oracle Clusterware and Oracle RAC are already installed on the system, Oracle Universal Installer displays the Specify Hardware Cluster Installation Mode screen. You must select **Local Installation** on this screen, unless you want to install Oracle RAC.

  See Also: *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for information on installing Oracle RAC

Selecting the Database Character Set

Oracle Database uses the database character set for:

- Data stored in SQL character datatypes (CHAR, VARCHAR2, CLOB, and LONG).
- Identifiers such as table names, column names, and PL/SQL variables.
- Stored SQL and PL/SQL source code, including text literals embedded in this code.

Once a database is created, changing its character set is usually very expensive in terms of time and resources. Such operation may require converting all character data by exporting the whole database and importing it back. Therefore, it is important that you carefully select the database character set already at installation time.

Oracle recommends Unicode AL32UTF8 as the database character set. Unicode is the universal character set that supports most of the currently spoken languages of the world. It also supports many historical scripts (alphabets). Unicode is the native encoding of many technologies, including Java, XML, XHTML, ECMAScript, and LDAP. Unicode is ideally suited for databases supporting the Internet and the global economy.

As AL32UTF8 is a multibyte character set, database operations on character data may be slightly slower when compared to single-byte database character sets, such as WE8MSWIN1252. Storage space requirements for text in most languages that use characters outside of the ASCII repertoire are higher in AL32UTF8 compared to legacy character sets supporting the language. Note that the increase in storage space concerns only character data and only data that is not in English. The universality and flexibility of Unicode usually outweighs these additional costs.

Legacy character sets should be considered when compatibility, storage requirements, or performance of text processing is critical and the database will ever support only a single group of languages. The database character set to be selected in such case is the character set of most clients connecting to this database.

The default character set suggested or used by Oracle Universal Installer and Database Configuration Assistant in this release is based on the language configuration of the operating system.

For most languages, the default character set is one of the Microsoft Windows character sets, for example WE8MSWIN1252, even though the database is not installed on Windows. This results from the assumption that most clients connecting to the database run under the Microsoft Windows operating system. As the database should be able to store all characters coming from the clients and Microsoft Windows character sets have richer character repertoire than the corresponding ISO 8859 character sets, the Microsoft Windows character sets are usually the better choice. For example, the EE8MSWIN1250 character set supports the Euro currency symbol and various smart quote characters, while the corresponding EE8ISO8859P2 character set does not support them. In any case, Oracle converts the data between the database character set and the client character sets, which are declared by the NLS_LANG settings.

The list of database character sets that is presented to you for selection by Oracle Universal Installer contains only the recommended character sets. Even though Oracle Database supports many more character sets, they are either deprecated or they are binary subsets of another recommended character set. For example, WE8DEC is a deprecated character set and US7ASCII and WE8ISO8859P1 are both binary subsets of WE8MSWIN1252.

If, for compatibility reasons, you need to create a database in one of the non-recommended character sets, choose the Custom installation type or choose the Advanced database configuration option. Database Configuration Assistant in the interactive mode will give you the opportunity to select any of the database character sets supported on Linux.

Installing the Sample Schemas

The Sample Schemas are not available in Basic Installation. There are two instances where the Sample Schemas are available:

- When a new database instance is created with the Database Configuration Assistant, the Sample Schemas can be installed. However, do not select Custom database. Sample Schemas are not available with a custom installation.
- When a new database instance is created with the Oracle Universal Installer, select either Enterprise Edition or Standard Edition, then select one of the two templates: General Purpose/Transaction Processing or Data Warehouse. The Sample Schemas can be installed. However, if you select the Advanced option on the Select Database Configuration screen, then the Sample Schemas are not available for installation.

See *Oracle Database Sample Schemas* for information about manually installing the Sample Schemas in an existing database.

Accessing the Installation Software

The Oracle Database software is available on installation media 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 installation media, if you have it.

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

- To copy the software to a hard disk, refer to "Copying the Software to the Hard Disk" on page 3-5
- To download the software from Oracle Technology Network, refer to "Downloading Oracle Software from the Oracle Technology Network Web Site" on page 3-4

Downloading Oracle Software from the Oracle Technology Network Web Site

This section describes how to download the installation archive files and extract them on to the 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 that you want to install.
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 `OraDB11g`, 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.
8. Extract the files in each directory that you just created.
9. After you have extracted the required installation files, refer to "Installing the Oracle Database Software" on page 3-7.

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. If the downloaded file has the `zip` extension, use the following command to extract the content:

   ```
   unzip file_name.zip
   ```

   If the downloaded file has the `cpio.gz` extension, use the following command:

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

   This command creates files with names similar to the following:

   ```
   filename.cpio
   ```

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

   ```
   $ cpio -idcmv < filename.cpio
   ```

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

   Some browsers uncompress files while downloading them, but leave the `.gz` file extension.

   For each file, this command creates a subdirectory named `Disk`*n*, where *n* is the disk number identified in the file name.

When you have extracted all of the required installation files, refer to "Installing the Oracle Database Software" on page 3-7.

Copying the Software to the Hard Disk

Before installing Oracle Database, you might want to copy the software to the hard disk. This enables the installation process to run a bit faster. Before copying the installation media content to the hard disk, you must mount the disk. The following sections describe to mount disk and copy its content to the hard disk.

Mounting Disks

On most Linux systems, the disk mounts automatically when you insert it into the installation media. If the disk does not mount automatically, then follow these steps to mount it:

1. If necessary, enter a command similar to one of the following to eject the currently mounted disk, then remove it from the drive:
   - Asianux, Oracle Enterprise Linux, and Red Hat Enterprise Linux:

     ```
     $ sudo eject /mnt/dvd
     ```

   - SUSE Linux Enterprise Server:

     ```
     # eject /media/dvd
     ```

In these examples, `/mnt/dvd` and `/media/dvd` are the mount point directories for the installation media.

2. Insert the appropriate installation media into the disk drive.
3. To verify if the disk is mounted automatically, enter one of the following commands depending on the platform:
   - Asianux, Oracle Enterprise Linux, and Red Hat Enterprise Linux:

     ```
     # ls /mnt/dvd
     ```

   - SUSE Linux Enterprise Server:

     ```
     # ls /media/dvd
     ```

4. Before running the following command, ensure that the `/mnt/dvd` directory exists on Red Hat Enterprise Linux. If not, create the `/mnt/dvd` as required, to mount the installation media.

   If this command fails to display the contents of the installation media, enter a command similar to the following to mount it, depending on the platform:

   - Asianux, Oracle Enterprise Linux, and Red Hat Enterprise Linux:

     ```
     # mount -t iso9660 /dev/dvd /mnt/dvd
     ```

   - SUSE Linux Enterprise Server:

     ```
     # mount -t iso9660 /dev/dvd /media/dvd
     ```

   In these examples, `/mnt/dvd` and `/media/dvd` are the mount point directories for the installation media.
5. If Oracle Universal Installer is displaying the Disk Location dialog box, enter the disk mount point directory path, for example:

   ```
   /mnt/dvd
   ```

To continue, go to one of the following sections:

- If you want to copy software to a hard disk, refer to "Copying the Oracle Database Software to a Hard Disk" on page 3-6.
- If you want to install the software from the installation media, refer to "Installing the Oracle Database Software" on page 3-7 .

Copying the Oracle Database Software to a Hard Disk

Note: If the system does not have a installation media, you can copy the software from the disk to a file system on another system, then either mount that file system using NFS, or use FTP to copy the files to the system where you want to install the software.

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

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

   ```
   $ mkdir OraDb11g
   ```

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

   ```
   $ cd OraDb11g
   ```

3. Mount the disk, if it is not already mounted.

   Some platforms automatically mount the disk when you insert it into the drive. If the disk does not mount automatically, refer to "Mounting Disks" section on page 3-5 for platform-specific information about mounting it.

4. Copy the contents of the mounted disk to the corresponding new subdirectory as follows:

   ```
   $ cp -R /directory_path OraDb11g
   ```

   In this example, `/directory_path` is the disk mount point directory.

5. If necessary, mount the next disk and repeat step 4.

Database Security Options

During installation, you are prompted to select a database security configuration. The Secure Configuration option configures the database with database auditing options, and password policy and expiration settings.

For new database installations, the default configuration for Oracle Database 11*g* Release 1 (11.1) includes the Secure Configuration option. If you want to disable these enhanced security controls, then you can check the **Disable security settings** box. Oracle Database is then installed with default options for Oracle Database 10*g* Release 2. After installation, you can change security settings by starting Database Configuration Assistant and modifying security settings. You can enable or disable auditing or password security settings, or revert to a previous security setting.

For database upgrades, the upgraded database retains your existing database security configuration, to ensure compatibility with existing applications. After installation, you can use Database Configuration Assistant to enable or disable the Secure Configuration option for testing.

Note:

- Oracle strongly recommends configuring your database with the Secure Configuration option either during installation, or after installation using Database Configuration Assistant.
- Database Vault is an enhanced Security feature. If it is installed with the database, then you cannot change the Secure Configuration using Database Configuration Assistant option.

Installing the Oracle Database Software

In most cases, you use the graphical user interface (GUI) provided by Oracle Universal Installer to install Oracle Database. The instructions in this section explain how to run the Oracle Universal Installer GUI to perform most database installations.

Note: If you run Oracle Universal Installer during the time that daily cron jobs run, then you may encounter unexplained installation problems if your cron job is performing cleanup, and temporary files are deleted before the installation is finished. Oracle recommends that you complete installation before daily cron jobs are run, or disable daily cron jobs that perform cleanup until after the installation is completed.

See Also:

- "Installing Automatic Storage Management" on page 3-14 if you want to install Oracle Database and use Automatic Storage Management
- Appendix A if you want to install Oracle Database by using the noninteractive installation method, without the GUI. This method is useful if you need to perform multiple installations of Oracle Database. This appendix covers other advanced installation topics as well.

Running 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. Log on as a member of the Administrators group to the computer on which you want to install Oracle components.
2. If you are installing the software from installation media, mount the disk if it is not already mounted.

   If the disk does not mount automatically, refer to "Mounting Disks" section on page 3-5 for platform-specific information about mounting it.

   Some platforms automatically mount the disk when you insert the installation media into the drive.
3. 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.

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

     ```
     $ /directory_path/runInstaller
     ```

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

     ```
     $ ./runInstaller
     ```

   If Oracle Universal Installer is not displayed, refer to "X Window Display Errors" on page G-1 for information about troubleshooting.

4. In the Welcome screen, the Basic Installation is selected by default. If you want to perform an advanced installation, then select Advanced Installation, and then answer the prompts as needed.

   See Also: "Oracle Database Installation Methods" on page 1-9 for more information on the Basic and Advanced installation methods

   The subsequent screens that appear, which are listed in the following table, depend on the installation method you have chosen. The order in which the screens appear depends on the options you select.

5. Use the following guidelines to complete the installation:

   - Do not install Oracle Database 11*g* Release 1 (11.1) software into an existing Oracle home.
   - Follow the instructions displayed on the Oracle Universal Installer screens. If you need additional information, click **Help**.

     See Also: "Reviewing Accounts and Passwords" on page 5-4 for details on password guidelines

   - Do not modify the Java Runtime Environment (JRE) except by using a patch provided by Oracle Support Services. Oracle Universal Installer automatically installs the Oracle-supplied version of the JRE. This version is required to run Oracle Universal Installer and several Oracle assistants.
   - If errors are displayed while installing the software, refer to Appendix G for information about troubleshooting.
   - If you chose an installation type that runs Oracle Database Configuration Assistant and Oracle Net Configuration Assistant in interactive mode, you must provide detailed information about configuring the database and network.

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

     Note: If you chose a default installation, Oracle Database Configuration Assistant and Oracle Net Configuration Assistant do not run interactively.

     ---

6. When the configuration assistant tasks are complete click finish, click **Exit**, then click **Yes** to exit from Oracle Universal Installer.

7. When Oracle Enterprise Manager Database Control opens in a Web browser, enter the user name and password you created during the installation.

   You can log in as `SYS`, `SYSTEM`, or `SYSMAN`. If you log in as `SYS`, then you must connect as `SYSDBA`. Enter the password you specified for the account during the installation.

8. 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
   ```

9. See Chapter 4 for information about tasks that you must complete after you have installed Oracle Database.

The following table lists the various screens displayed during database installation:

> **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.

Screen	Action
Select a Product to Install	This screen enables you to install any one of the following products: ■ Oracle Database 11g ■ Oracle Client ■ Oracle Clusterware Click **Next.**
Select Installation Method	Select one of the following, then click **Next**: ■ Basic Installation: This installation method is selected by default. It lets you quickly install Oracle Database using minimal input. It installs the software and optionally creates a general-purpose database using the information that you specify on this screen. ■ Advanced Installation: Lets you perform more complex installations, such as creating individual passwords for different accounts, creating specific types of starter databases (for example, for transaction processing or data warehouse systems), using different language groups, specifying e-mail notifications, and so on.
Specify Inventory Directory and Credentials	This screen is displayed only during the first installation of Oracle products on a system. Specify the full path of the Oracle Inventory directory. Click Next. The next screen contains the operating system group selected by default.
Select Installation Type	Select Enterprise Edition, Standard Edition, or Custom. You can also specify language translations to be installed by clicking on Product Languages. Click **Next**. Note: This screen is available only with Advanced Installation.
Install Location	The Oracle base path appears by default. You can change the path based on your requirement. In the **Software Location** section, accept the default values or enter the Oracle home name and directory path in which you want to install Oracle components. The directory path should not contain spaces. Click **Next**. Note: This screen is available only with Advanced Installation.

Screen	Action
Available Product Components	If you selected Custom for the Installation Type, this screen is displayed. Select the components to be installed from the list and click **Next**. To learn more about each component, place the mouse over the component name. Note: This screen is available only with Advanced Installation.
Product-specific Prerequisite Checks	This screen checks that the system meets the minimum requirements for the installation. Correct any errors that Oracle Universal Installer may have found, and then click **Next**.
Upgrade an Existing Database	This screen is displayed if you have a previous updatable version of Oracle Database or Automatic Storage Management installed. For in-place database installations where Automatic Storage Management is running, Automatic Storage Management is upgraded automatically. Click **Yes** if you want to upgrade or **No** if not. If you click Yes, the Summary screen is displayed. For more information about upgrades, refer *Oracle Database Upgrade Guide*.
Select Configuration Option	Select one of the following: ■ Create a database**:** Select this option if you are creating a database of the following types: General purpose / Transaction processing, and Data warehousing. The Advanced option lets you perform a custom installation. ■ Configure Automatic Storage Management: Select this option to create an Automatic Storage Management instance only. To create an Automatic Storage Management instance, you must provide an Automatic Storage Management `SYS` Password. After you provide this password, Oracle Universal Installer lets you create an Automatic Storage Management disk group. After you complete this Oracle Universal Installer session, you can run it again to install and configure one or more Oracle databases that use Automatic Storage Management. ■ Install database Software only: Select this option to install the database software only. This option does not create a database or configure Automatic Storage Management.
Select Automatic Storage Management Option	If you selected Configure Automatic Storage Management from the Select Configuration Option screen, and if you have Oracle Enterprise Manager 10*g* Grid Control installed, then this screen is displayed. Select **Yes** or **No**, depending on the requirement to use Grid Control to manage Automatic Storage Management. If you select **Yes**, then select from the list of Enterprise Management agents to use.

Screen	Action
Configure Automatic Storage Management	If you selected Configure Automatic Storage Management from the Select Configuration Option screen, this screen is displayed. Enter the disk group name. The disk group list shows both candidate and member disks; you can click **Show Candidates** or **Show All** to filter their display. Then, select the redundancy level and member disks for the disk group. For Redundancy Level, choose one of the following options. If you do not choose a redundancy level, the disk group defaults to normal redundancy. ■ High: With this option, the contents of the disk group are three-way mirrored by default. To create a disk group with high redundancy, you must specify at least three failure groups (a minimum of three devices). ■ Normal: In a normal redundancy level, by default, the data files of the disk group are two-way mirrored and the control files are three-way mirrored. You can choose to create certain files that are three-way mirrored or not mirrored. To create a disk group with normal redundancy, you must specify at least two failure groups (a minimum of two devices) for two-way mirroring. ■ External: If you select this option, Automatic Storage Management does not mirror the contents of the disk group. Choose this redundancy level when the disk group contains devices, such as RAID devices, that provide their own data protection; or the use of the database does not require uninterrupted access to data, for example, in a development environment where you have a suitable backup strategy.
Select Database Configuration	Select the database configuration that best meets the requirements: General Purpose / Transaction Processing, Data Warehouse, or Advanced. See the online Help provided by either Oracle Universal Installer or Oracle Database Configuration Assistant for a description of these preconfigured database types. Click **Next**.

Screen	Action
Specify Database Configuration Options	Specify the following information, then click **Next**: Database Naming Specify the Global Database Name using the following syntax: *database_name*.*domain* where: ■ *database_name* is the name of the database. It can contain no more than 30 characters (alphanumeric, underscore (_), dollar ($), and pound (#)). ■ *domain* is the domain used for the database. It can contain no more than 128 characters (alphanumeric, underscore (_), and pound (#)), inclusive of all periods. For example: sales.us.example.com When you enter the Global Database Name, Oracle Universal Installer automatically populates the SID field with the database name, but you can change this SID to another name. The SID can have no more than 64 characters (alphanumeric, dollar ($), and pound (#)).
Specify Database Config Details	Specify the following configuration details, then click **Next**: Memory Specify the amount of memory (RAM) you want to allocate in the **Percentage** field. If you install the database software only, then you can click **Show Memory Distribution** to check the memory usage by the various processes running on the system. Character Set Determine how character data is encoded in the database. The default is based on the operating system language. Select Unicode (AL32UTF8) to store multiple languages. **See Also:** ■ "Selecting the Database Character Set" ■ *Oracle Database Globalization Support Guide* for information on choosing a character set. Security To disable the default enhanced security controls, you can check the Disable security settings box. Oracle Database is then installed with default options for Oracle Database 10*g* Release 2. Sample Schema You can specify if you want to create Oracle Database with or without sample schemas.
Select Database Management Option	Select one of the following, then click **Next**: ■ Use Grid Control for Database Management if you have Oracle Enterprise Manager installed. ■ Use Database Control for Database Management. Optionally, select **Enable Email Notifications** and then enter the outgoing SMTP server and e-mail address.

Screen	Action
Specify Database Storage Option	Select one of the following, then click **Next**. ■ File System: Specify the database file location. ■ Automatic Storage Management
Specify Backup and Recovery Options	Select one of the following, then click **Next**. ■ Do not enable Automated backups ■ Enable Automated Backups: Specify the recovery area storage location and backup job credentials
Specify Database Schema Passwords	Enter and confirm passwords for the privileged database accounts, then click **Next**. **Note:** Optionally, you can use the same password for all accounts. However, Oracle recommends that you specify a different password for each account. You must remember the passwords that you specify. Refer to "Unlocking and Changing Passwords" on page 5-8 for information on password guidelines.
Privileged Operating System Groups	This screen is displayed only during the first installation of Oracle products on a system. The groups are selected by default. Click **Next**.
Oracle Configuration Manager Registration	Enter the Customer Identification Number, Oracle*Metalink* User Name, Country code, and Click **Next**. The new screen prompts you to accept the license agreement. Click **Accept license Agreement** to accept the agreement. If you decline this agreement, then Oracle Configuration Manager is installed but not configured.
Summary	Review the information displayed on this screen. 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 at the end of Database Configuration Assistant 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	The configuration assistants configure several Web-based applications, including Oracle Enterprise Manager Database Control. 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: `$ORACLE_HOME/install/portlist.ini` To exit from Oracle Universal Installer, click **Exit** and then click **Yes**.

Installing Automatic Storage Management

Follow the procedures in this section to install and configure Automatic Storage Management, and then to install Oracle Database so that it can use Automatic Storage Management. If you do not plan to use Automatic Storage Management, then follow

the procedure given in "Installing the Oracle Database Software" on page 3-7 to install Oracle Database.

This section covers the following topics:

- Step 1: Reviewing Automatic Storage Management Installation Considerations
- Step 2: Installing the Automatic Storage Management Instance and configuring Disk Groups
- Step 3: Installing Oracle Database to Use Automatic Storage Management
- Step 4: Testing the Automatic Storage Management Installation

Step 1: Reviewing Automatic Storage Management Installation Considerations

Apply these guidelines when you install Automatic Storage Management:

- You must complete the steps in "Preparing Disk Groups for an Automatic Storage Management Installation" on page 2-32 to prepare a disk partition to use for the Automatic Storage Management disk groups.
- Oracle recommends that you install Automatic Storage Management in its own Oracle home, regardless of whether you plan to only have one or multiple database instances. Installing Automatic Storage Management in its own Oracle home helps ensure higher availability and manageability.

  With separate Oracle homes, you can upgrade Automatic Storage Management and databases independently, and you can remove database software without impacting the Automatic Storage Management instance. Ensure that the Automatic Storage Management version is the same or later than the Oracle Database version.

  If an Automatic Storage Management installation does not already exist and you select the Oracle Universal Installer option to install and configure Automatic Storage Management only, then Oracle Universal Installer installs Automatic Storage Management in its own Oracle home.
- Each computer that has one or more Oracle Database instances that will use Automatic Storage Management must have one Automatic Storage Management instance. For example, if a computer has two Oracle Database instances that use Automatic Storage Management, you need only one Automatic Storage Management instance for that computer, to manage the two database instances that use Automatic Storage Management.
- When you install Automatic Storage Management, Oracle Database Configuration Assistant creates a separate server parameter file (`SPFILE`) and password file for the Automatic Storage Management instance.

Step 2: Installing the Automatic Storage Management Instance and configuring Disk Groups

The following steps explain how to create an Automatic Storage Management instance and a disk group for storing Oracle Database files. You can create multiple disk groups for the Automatic Storage Management instance to manage, if you want. If you plan to use Automatic Storage Management for backup and recovery operations, then Oracle recommends that you create a separate disk group for this purpose.

To install an Automatic Storage Management instance and configure its disk groups:

1. If you are installing the software from disk, mount the first installation media if it is not already mounted.

   Some platforms automatically mount the disk when you insert the installation media into the drive.

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 installation media, enter commands similar to the following, where *`directory_path`* is the path of the `database` directory on the installation media:

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

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

     ```
     $ ./runInstaller
     ```

   If Oracle Universal Installer does not appear, refer to the "X Window Display Errors" section on page G-1 for information about troubleshooting.

3. The Select a Product to Install screen enables you to install any one of the following products:

   - Oracle Database 11*g*
   - Oracle Client
   - Oracle Clusterware

   See Also: "Running Oracle Universal Installer" on page 3-8 for a detailed description of the screens used in this procedure

4. On the Select Installation Type screen, select either **Enterprise Edition**, **Standard Edition**, or **Custom**, and then click **Next**.

5. On the Install location screen, the Oracle base path appears by default. You can change the path based on your requirement. In the **Software Location** section, accept the default values or enter an Automatic Storage Management-specific name and directory location for the Automatic Storage Management instance and Click **Next**.

   For example, you could change name to `OraDB11g+ASM` and the directory location to the following:

   ```
   /u01/app/oracle/product/11.1.0/asm
   ```

6. On the Product-Specific Prerequisite Checks screen, check that the requirements have been met and then click **Next**.

7. On the Select Configuration Option screen, select **Configure Automatic Storage Management (ASM)** and then specify and confirm the Automatic Storage Management SYS password. Then, click **Next**.

8. On the Configure Automatic Storage Management screen, enter the following settings:

   Note: This screen lets you create the disk groups to use with the Automatic Storage Management instance. You must have an available partition in order to create disk groups.

   - **Disk Group Name**: Enter a name for the disk group.
   - Redundancy: Select one of the following choices to set the redundancy level for the disks within the disk group. If you do not specify a redundancy level, the disk group defaults to normal redundancy.
     - **High**: The contents of the disk group are three-way mirrored by default. To create a disk group with high redundancy, you must specify at least three failure groups (a minimum of three devices).
     - **Normal**: By default, the data files of the disk group are two-way mirrored and the control files are three-way mirrored. You can choose to create certain files that are three-way mirrored or not mirrored. To create a disk group with normal redundancy, you must specify at least two failure groups (a minimum of two devices) for two-way mirroring.
     - **External**: Automatic Storage Management does not mirror the contents of the disk group. Choose this redundancy level when 1) the disk group contains devices, such as RAID devices, that provide their own data protection; or 2) the use of the database does not require uninterrupted access to data, for example, in a development environment where you have a suitable backup strategy.
   - **Add Disks**: Enter `/dev/rdsk` on as the disk discovery path. You can also click **Disk Discovery Path** and enter the required disk discovery path in the dialog box and click **OK**.

9. Click **Next**.
10. On the Install screen, check the installed contents, and then click **Install**.
11. To create another disk group for this instance, run Oracle Database Control Assistant from the `$ORACLE_HOME/bin` directory manually, and select the **Configure Automatic Storage Management** option.

At this stage, subsequent databases that you create are able to use Automatic Storage Management. If you have databases that were created before you installed Automatic Storage Management, then you now can migrate them to Automatic Storage Management by using the Enterprise Manager Migrate Database Wizard. This wizard is available in Enterprise Manager Grid Control or Database Control. Alternatively, you can use Oracle Database Recovery Manager (RMAN) to perform the migration.

See Also:

- Enterprise Manager Migrate Database Wizard online Help instructions on how to migrate an existing Oracle Database to Automatic Storage Management
- *Oracle Database Backup and Recovery User's Guide* for information about migrating an existing Oracle Database to Automatic Storage Management using Oracle Database Recovery Manager

Step 3: Installing Oracle Database to Use Automatic Storage Management

After you have created the Automatic Storage Management instance and Automatic Storage Management disk groups, you are ready to create a database instance that can use Automatic Storage Management.

To create a database instance to use with Automatic Storage Management:

1. Start Oracle Universal Installer.
2. The Select a Product to Install screen enables you to install any one of the following products:
   - Oracle Database 11*g*
   - Oracle Client
   - Oracle Clusterware
3. On the Select Installation Type screen, select one of the installation types, and then click **Next**.
4. On the Install location screen, the Oracle base path appears by default. You can change the path based on your requirement. In the **Software Location** section, accept the default values or select a different Oracle home from the home used for Automatic Storage Management.
5. If you selected the Custom installation type, then select from the products to install.
6. On the Product-Specific Prerequisite Checks screen, check that the requirements have been met and then click **Next**.
7. On the Select Configuration Option screen, select **Create a Database**.
8. On the Select Database Configuration screen, select from the database types displayed and click **Next**.
9. On the Specify Database Configuration Options screen, enter the following settings and then click **Next**.
   - Specify the Global Database Name using the following syntax:

     database_name.domain

     where:
     - *database_name* with no more than 30 characters (alphanumeric, underscore (_), dollar ($), and pound (#)).
     - *domain* name with no more than 128 characters (alphanumeric, underscore (_), and pound (#)), inclusive of all periods.
   - Specify the SID with less than 64 characters (alphanumeric, dollar ($), and pound (#)).
10. On the Specify Database Config Details screen, enter the following settings, then click **Next**:
    - Memory
    - Character Sets
    - Security
    - Sample Schema

See Also: "Running Oracle Universal Installer" on page 8 for further information about these fields.

11. On the Select Database Management Option screen, select either **Use Grid Control for Database Management** if you have Oracle Enterprise Manager installed, or if you do not have Enterprise Manager, select **Use Database Control for Database Management**. Optionally, select **Enable Email Notifications** and then enter the outgoing SMTP server and e-mail address. Then, click **Next**.

    After you complete the installation, you can use either of these utilities to manage the Automatic Storage Management instance.

12. On the Specify Database Storage Option screen, select **Automatic Storage Management (ASM)** and click **Next**.

13. On the Specify Backup and Recovery Options screen, perform the following actions:

    - **Enable Automated Backups**: Select this option, and then select **Automatic Storage Management**.
    - **Backup Job Credentials**: Enter the user name and password of the person responsible for managing backups.

14. Click **Next**.

15. On the Select Automatic Storage Management Disk Group screen, select the Automatic Storage Management disk group that you created in "Step 2: Installing the Automatic Storage Management Instance and configuring Disk Groups" on page 3-15 for recovery and backups.

    If the Automatic Storage Management disks that you select do not provide enough space, then the Configure Storage Management screen is displayed so that you can select additional disks as needed. As you select the disks, the adjusted sizes are displayed in the Required Storage Space region. Ideally, the **Additional Space Needed** value should be a negative number.

16. Click **Next**.

17. On the Specify Database Schema Passwords screen, enter and confirm passwords for the privileged database accounts, then click **Next**.

18. On the Oracle Configuration Manager screen, enter the Customer Identification Number, Metalink User Name, Country code, and Click **Next**. The new screen prompts you to accept the license agreement. Click **Accept license Agreement** to accept the agreement. However, if you decline this agreement, then Oracle Configuration Manager is installed but not configured.

19. On the Summary screen, check that the contents to be installed are correct, and then click **Install**.

Step 4: Testing the Automatic Storage Management Installation

To test the Automatic Storage Management installation, try logging in by using the `asmcmd` command-line utility, which lets you manage Automatic Storage Management disk group files and directories. To do this:

1. Open a shell window, and temporarily set the `ORACLE_SID` and `ORACLE_HOME` environment variables to specify the appropriate values for the Automatic Storage Management instance that you want to use.

For example, if the Automatic Storage Management SID, which is named +ASM and is located in the asm subdirectory of the ORACLE_BASE directory, then enter the following commands to create the required settings:

- Bourne, Bash, or Korn shell:

```
$ ORACLE_SID=+ASM
$ export ORACLE_SID
$ ORACLE_HOME=/u01/app/oracle/product/11.1.0/asm
$ export ORACLE_HOME
```

- C shell:

```
% setenv ORACLE_SID +ASM
% setenv ORACLE_HOME /u01/app/oracle/product/11.1.0/asm
```

2. By using SQL*Plus, connect to the Automatic Storage Management instance as the SYS user with SYSASM privilege and start the instance if necessary:

```
# $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSASM
Enter password: SYS_password
SQL> STARTUP
```

3. Enter the following command to view the existing disk groups, their redundancy level, and the amount of free disk space in each one:

```
SQL> SELECT NAME,TYPE,TOTAL_MB,FREE_MB FROM V$ASM_DISKGROUP;
```

See Also:

- *Oracle Database Utilities* for more information about asmcmd
- "Managing Automatic Storage Management" on page 5-3 for information about other tools that you can use to manage Automatic Storage Management
- *Oracle Database Administrator's Guide* for a more detailed description of Automatic Storage Management

Installing Oracle Database Examples

If you plan to use the following products or features, then download and install the products from the Oracle Database Examples media:

- Oracle JDBC Development Drivers
- Oracle Database Examples
- Oracle Text Knowledge Base
- Various Oracle product demonstrations

You must install the Sample Schemas in order to use Oracle Database Examples.

See Also: ■

- *Oracle Database Examples Installation Guide*.for more information on various Oracle product demonstrations.
- "Installing the Sample Schemas" on page 3-3

4

Oracle Database Postinstallation Tasks

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

- Required Postinstallation Tasks
- Recommended Postinstallation Tasks
- Required Product-Specific Postinstallation Tasks
- Postinstallation tasks for SQL Developer
- Postinstallation Tasks for Oracle Application Express
- Postinstallation Database Configuration for Oracle Configuration Manager

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 install 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 Linux and UNIX*, *Oracle Database Administrator's Guide* 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 Oracle*MetaLink* Web site for required patches for the installation.

To download required patches:

1. Use a Web browser to view the Oracle*MetaLink* Web site:

   `https://metalink.oracle.com`

2. Log in to Oracle*MetaLink*.

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

3. On the main Oracle*MetaLink* page, click **Patches and Updates**.
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.
6. Find the latest patch set for Oracle Database using Oracle*MetaLink*.
7. From the list of available patches, select a patch to download.

   Patch sets for Oracle databases are identified as *x.x.x* PATCH SET FOR ORACLE DATABASE SERVER.
8. Review the README file before proceeding with the download.

   Each patch has a README file with installation requirements and instructions. Some patches install with Oracle Universal Installer; others require special procedures. Oracle recommends that you always read the README file before proceeding.
9. Download and install the patch.

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
- Direct NFS Client

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 `utlrp.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.

> **See Also:** *Oracle Database Upgrade Guide* for more information about database upgrade.

To run the `utlrp.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 `utlrp.sql` script:
   - Bourne, Bash, or Korn shell:

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

   - 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> @?/rdbms/admin/utlrp.sql
   ```

Setting Up User Accounts

For information about setting up additional user accounts, refer to *Oracle Database Administrator's Reference for Linux and UNIX*.

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 declares the character set of the client, which is the character set of data entered or displayed by an Oracle client program, such as SQL*Plus.

> **See Also:** Appendix F, "Configuring Oracle Database Globalization Support" for more information about the `NLS_LANG` environment variable

Generating the Client Static Library

The client static library (`libclntst11.a`) is not generated during installation. If you want to link the applications to the client static library, 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:
   - Bourne, Bash, or Korn shell:

     ```
     $ ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1
     $ export ORACLE_HOME
     ```

   - C shell:

     ```
     % setenv ORACLE_HOME /u01/app/oracle/product/11.1.0/db_1
     ```

3. Enter the following command:

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

Direct NFS Client

Network-attached storage (NAS) systems use NFS to access data. You can store data files on a supported NFS system.

With Oracle Database 11*g*, instead of using the operating system kernel NFS client, you can configure Oracle Database to access NFS V3 servers directly using an Oracle internal Direct NFS client.

If Oracle Database is unable to open an NFS server using Direct NFS, then Oracle Database uses the platform operating system kernel NFS client. In this case, the kernel NFS mount options must be set up as defined in "Checking NFS Buffer Size Parameters" on page 4-7. Additionally, an informational message will be logged into the Oracle alert and trace files indicating that Direct NFS could not be established.

The Oracle files resident on the NFS server that are served by the Direct NFS Client are also accessible through the operating system kernel NFS client. The usual considerations for maintaining integrity of the Oracle files apply in this situation.

Some NFS file servers require NFS clients to connect using reserved ports. If your filer is running with reserved port checking, then you must disable it for Direct NFS to operate. To disable reserved port checking, consult your NFS file server documentation.

Direct NFS can use up to four network paths defined for an NFS server. The Direct NFS client performs load balancing across all specified paths. If a specified path fails, then Direct NFS reissues I/Os over any remaining paths.

Use the following views for Direct NFS management:

- **v$dnfs_servers:** Shows a table of servers accessed using Direct NFS.
- **v$dnfs_files:** Shows a table of files currently open using Direct NFS.
- **v$dnfs_channels:** Shows a table of open network paths (or channels) to servers for which Direct NFS is providing files.
- **v$dnfs_stats:** Shows a table of performance statistics for Direct NFS.

The following sections elaborate on enabling, disabling, checking the buffer size for a Direct NFS Client:

Enabling Direct NFS Client

By default Direct NFS will attempt to serve mount entries found in `/etc/filesystems`. No other configuration is required. You can use `oranfstab` to specify additional Oracle specific options to Direct NFS. For example, you can use `oranfstab` to specify additional paths for a mount point.

Additonally, a new Oracle specific file `oranfstab` can be added to either `/etc` or to `$ORACLE_HOME/dbs`. When `oranfstab` is placed in `$ORACLE_HOME/dbs`, its entries are specific to a single database. However, when `oranfstab` is placed in `/etc`, then it is global to all Oracle databases, and hence can contain mount points for all Oracle databases.

Direct NFS determines mount point settings to NFS storage devices based on the configurations in `/etc/mtab`. Direct NFS looks for the mount point entries in the following order:

1. `$ORACLE_HOME/dbs/oranfstab`
2. `/etc/oranfstab`
3. `/etc/mtab`

It uses the first matched entry as the mount point.

In all cases, Oracle requires that mount points be mounted by the kernel NFS system even when being served through Direct NFS. Oracle verifies kernel NFS mounts by cross-checking entries in `oranfstab` with operating system NFS mount points. If a mismatch exists, then Direct NFS logs an informational message, and does not serve the NFS server.

Complete the following procedure to enable Direct NFS:

1. You can optionally create an `oranfstab` file with the following attributes for each NFS server to be accessed using Direct NFS:
   - Server: The NFS server name.
   - Path: Up to four network paths to the NFS server, specified either by IP address, or by name, as displayed using the `ifconfig` command on the filer.
   - Local: Up to four local paths on the database host, specified by IP address or by name, as displayed using the `ifconfig` command run on the database host.
   - Export: The exported path from the NFS server.
   - Mount: The local mount point for the NFS server.
   - Dontroute: Specifies that outgoing messages should not be routed by the operating system, but sent using the IP address they are bound to.

Note:

- On Unix platforms, the location of the `oranfstab` file is `$ORACLE_HOME/dbs`.
- The parameters `local` and `dontroute` are available from patchset 11.1.0.7 onwards.

The examples below show different possible NFS server entries in `oranfstab`. A single `oranfstab` can have more than one NFS server entries.

- The following example uses both local and path. Since they are in different subnets, we do not have to specify `dontroute`:

```
server: MyDataServer1
local: 132.34.35.10
path: 132.34.35.12
local: 132.44.35.10
path: 132.44.35.12
export: /vol/oradata1 mount: /mnt/oradata1
```

- The following example shows local and path in the same subnet. `dontroute` is specified in this case:

```
server: MyDataServer2
local: 132.40.35.12
path: 132.40.45.12
local: 132.40.35.13
path: 132.40.45.13
dontroute
export: /vol/oradata2 mount: /mnt/oradata2
```

- This example uses names instead of IP addresses. Also, note that you can have more than one export:

```
server: MyDataServer3
local: LocalPath1
path: NfsPath1
local: LocalPath2
path: NfsPath2
local: LocalPath3
path: NfsPath3
local: LocalPath4
path: NfsPath4
dontroute
export: /vol/oradata3 mount: /mnt/oradata3
export: /vol/oradata4 mount: /mnt/oradata4
export: /vol/oradata5 mount: /mnt/oradata5
export: /vol/oradata6 mount: /mnt/oradata6
```

2. Oracle Database uses an ODM library, `libnfsodm11.so`, to enable Direct NFS. To replace the standard ODM library, `$ORACLE_HOME/lib/libodm11.so`, with the ODM NFS library, `libnfsodm11.so`, complete the following steps:

   - Change directory to `$ORACLE_HOME/lib`.
   - Enter the following commands:

```
cp libodm11.so libodm11.so_stub
ln -s libnfsodm11.so libodm11.so
```

Disabling Direct NFS Client

Use one of the following methods to disable the Direct NFS client:

- Remove the `oranfstab` file.
- Restore the stub `libodm11.so` file by reversing the process you completed in "Enabling Direct NFS Client" on page 4-5.
- Remove the specific NFS server or export paths in the `oranfstab` file.

> **Note:** If you remove an NFS path that Oracle Database is using, then you must restart the database for the change to be effective.

Checking NFS Buffer Size Parameters

If you are using NFS, then you must set the values for the NFS buffer size parameters rsize and wsize to at least 16384. Oracle recommends that you use the value 32768.

Direct NFS will issue writes at wtmax granularity to the NFS server. Direct NFS will not serve an NFS server with a wtmax less than 32768.

For example, if you decide to use rsize and wsize buffer settings with the value32768, then update the `/etc/vfstab` file on each node with an entry similar to the following:

```
nfs_server:/vol/DATA/oradata /home/oracle/netapp nfs\
rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,vers=3,timeo=600
```

> **Note:** Refer to your storage vendor documentation for additional information about mount options.

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
- Configuring Oracle Database Vault
- Configuring Oracle Messaging Gateway
- Configuring Oracle Precompilers
- Configuring Secure Sockets Layer
- Installing Oracle Text Supplied Knowledge Bases

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

Configuring Oracle Net Services

If you have an earlier release of Oracle software installed on this system, you might want to copy information from the Oracle Net `tnsnames.ora` and `listener.ora` configuration files from the earlier release to the corresponding files for the new release.

The following sections describe about how to configure the Oracle Net Services:

- Modifying the `listener.ora` File
- Modifying the `tnsnames.ora` File

> **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.

Modifying the `listener.ora` File

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

To use the listener from the current release, you may need to copy static service information from the `listener.ora` file from the earlier 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 earlier 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, 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, 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

Configuring Oracle Database Vault

If you have installed Oracle Database Vault, then you must register it in a database. Ensure that you create the Database Vault Owner and, optionally, Database Vault Account Manager administrative accounts before you can use it.

See Also: *Oracle Database Vault Administrator's Guide* for more information on registering Oracle Database Vault.

Configuring Oracle Messaging Gateway

To configure Oracle Messaging Gateway, refer to the section about Messaging Gateway in *Oracle Streams Advanced Queuing User's Guide*. 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 = "LD_LIBRARY_PATH=/oracle_home/jdk/jre/lib/i386:/oracle_home/jdk \
/jre/lib/i386/server:/oracle_home/lib")
    (ORACLE_HOME = oracle_home)
    (PROGRAM = extproc)
  )
```

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.
- `extproc` is the external procedure agent executable file

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

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PLSExtProc)
      (ORACLE_HOME = /u01/app/oracle/product/11.1.0/db_1)
      (PROGRAM = extproc)
    )
    (SID_DESC =
      (SID_NAME = mgwextproc)
(ENVS = "LD_LIBRARY_PATH =/u01/app/oracle/product/11.1.0/db_1/jdk/jre/ \
lib/i386:/u01/app/oracle/product/11.1.0/db_1/jdk/jre/lib/i386/server: \
/u01/app/oracle/product/11.1.0/db_1/lib")
       (ORACLE_HOME = /u01/app/oracle/product/11.1.0/db_1)
       (PROGRAM = extproc)
```

```
        )
    )
```

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)))
```

   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
JRE runtime	`$ORACLE_HOME/jdk/jre/lib/rt.jar`
Oracle JDBC	`$ORACLE_HOME/jdbc/lib/ojdbc5.jar`
Oracle internationalization	`$ORACLE_HOME/jlib/orai18n.jar`
SQLJ	`$ORACLE_HOME/sqlj/lib/runtime12.jar`
JMS Interface	`$ORACLE_HOME/rdbms/jlib/jmscommon.jar`
Oracle JMS implementation	`$ORACLE_HOME/rdbms/jlib/aqapi.jar`
Java Transaction API	`$ORACLE_HOME/jlib/jta.jar`

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

Configuring Oracle Precompilers

This section describes postinstallation tasks for Oracle precompilers:

- Configuring Pro*C/C++
- Configuring Pro*FORTRAN

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.

Table 4–1 shows the default directories and the appropriate command to verify the path setting of the compiler.

Table 4–1 C/C++ Compiler Directory

Path	Command
`/usr/bin`	`$ which gcc`
`/opt/intel_cce_80/bin/icc`	`$ which icc`

Configuring Pro*FORTRAN

Verify that the PATH environment variable setting includes the directory that contains the FORTRAN compiler executable. You can verify the path setting by using the `which xlf` command. The path for the FORTRAN executable is `/usr/bin`.

Configuring Secure Sockets Layer

Oracle highly recommends you configure and use a Secure Sockets Layer (SSL) to ensure that passwords and other sensitive data are not transmitted in clear text in HTTP requests.

See Also:

- "Using SSL" and "Enabling SSL" in *Oracle Database Advanced Security Administrator's Guide* for more information on configuring and using SSL
- "SSL Usage Issues" in *Oracle Database Advanced Security Administrator's Guide* for more information on SSL usage issues

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).

See Also:

- *Oracle Database Examples Installation Guide*
- *Oracle Text Reference* for information about creating and extending knowledge bases, such as extending the supplied knowledge bases to accommodate your requirements, or creating your own knowledge bases in languages other than English and French

Postinstallation tasks for SQL Developer

This section describes tasks that you need to complete after you install the software:

Migrating User Settings from Release 1.0

The first time you start SQL Developer after installing it or after adding any extensions, you are asked if you want to migrate your user settings from a previous release. (This occurs regardless of whether there was a previous release on your system.)

Note: Migration of user settings is supported only from SQL Developer release 1.0 to release 1.1. It is *not* supported for migration from a pre-release version of 1.1 to release 1.1.

These settings refer to database connections, reports, and certain SQL Developer user preferences that you set in a previous version by clicking **Tools** and then **Preferences**. However, some user preferences are not saved, and you must re-specify these using the new release.

To migrate user settings from SQL Developer release 1.0:

1. Unzip the release 1.1 kit into an empty directory (folder). Do not delete or overwrite the directory into which you unzipped the release 1.0 kit.
2. When you start SQL Developer release 1.1, click **Yes** when asked if you want to migrate settings from a previous release.
3. In the dialog box that is displayed, do *not* accept the default location for the settings. Instead, specify the location of your release 1.0 settings, which might be a folder whose path ends with `sqldeveloper/jdev/system`.

See Also: "Migrating Information from Previous Releases" on page 4-12 for more information

Migrating Information from Previous Releases

If you have used a previous release of SQL Developer or a pre-release version of the current release, you may want to preserve database connections that you have been using. To preserve database connections, save your existing database connections in an XML file. To save the connections, right-click the Connections node in the Connections Navigator and select **Export Connections**. After you complete the installation described in this guide, you can use those connections by right-clicking the Connections node in the Connections Navigator and selecting **Import Connections**

If you want to use any user-defined reports or the SQL history from a previous version, see "Location of User-Related Information" on page 4-13 for information about where these are located. If you want to use any user-defined reports or the SQL history from release 1.0 with both releases 1.0 and 1.1, you must save them before using release 1.1, because release 1.1 modifies the files to a format that is incompatible with release 1.0.

SQL Developer preferences (specified by clicking Tools and then Preferences) from a pre-release version of the current release cannot currently be saved and reused; you must re-specify any desired preferences.

Location of User-Related Information

SQL Developer stores user-related information in several places, with the specific location depending on the operating system and certain environment specifications. User-related information includes user-defined reports, user-defined snippets, SQL Worksheet history, and SQL Developer user preferences.

The user-related information is stored outside the SQL Developer installation directory hierarchy, so that it is preserved if you delete that directory and install a new version. This information is stored in or under the `SQLDEVELOPER_USER_DIR` location, if defined; otherwise as indicated in the following table.

The table shows the typical default locations (under a directory or in a file) for specific types of resources on different operating systems. (Note the period in the name of any directory or folder named `.sqldeveloper`.)

Table 4–2 Default Locations for User-Related Information

Resource Type	Linux
User-defined reports	~/.sqldeveloper/UserReports.xml
User-defined snippets	~/.sqldeveloper/UserSnippets.xml
SQL history	~/.sqldeveloper/system/
SQL Worksheet archive files	~/.sqldeveloper/tmp/
SQL Developer user preferences	~/.sqldeveloper/system/

SQL Worksheet archive files contain SQL statements that you have entered. These files begin with sqldev and then have a random number (for example, `sqldev14356.sql`). If you close SQL Developer with a SQL Worksheet open that contains statements, then you will be prompted to save these files.

To specify a nondefault `SQLDEVELOPER_USER_DIR` location, do either of the following:

- Set the `SQLDEVELOPER_USER_DIR` environment variable to specify another directory path.
- Edit the *`sqldeveloper_install`*`\sqldeveloper\sqldeveloper\bin\sqldeveloper.conf` file and substitute the desired directory path for `SQLDEVELOPER_USER_DIR` in the following line:

  ```
  SetUserHomeVariable SQLDEVELOPER_USER_DIR
  ```

If you want to prevent other users from accessing your user-specific SQL Developer information, you must ensure that the appropriate permissions are set on the directory where that information is stored or on a directory preceding it in the path hierarchy. For example, you may want to ensure that the `~/.sqldeveloper` directory is not world-readable.

Postinstallation Tasks for Oracle Application Express

This section describes the following tasks that you need to complete after you install the software:

- Restarting Processes
- Choosing an HTTP Server

- Configuring the Embedded PL/SQL Gateway
- Copying the Images Directory
- Configuring Oracle HTTP Server in a New Installation
- Enabling Network Services in Oracle Database 11g
- Running Oracle Application Express in Other Languages
- Managing JOB_QUEUE_PROCESSES
- Obfuscating PlsqlDatabasePassword Parameter
- Logging In to Oracle Application Express
- Patching Oracle Application Express 3.0

Note: Within the context of this document, the Oracle HTTP Server home directory (`ORACLE_HTTPSERVER_HOME`) is the location where Oracle HTTP Server is installed.

Restarting Processes

After you install Oracle Application Express, you need to restart the processes that you stopped before you began the installation, such as listener and other processes. In addition, restart Oracle HTTP Server.

Choosing an HTTP Server

In order to run, Oracle Application Express must have access to either the embedded PL/SQL gateway or Oracle HTTP Server and `mod_plsql`.

Topics in this section include:

- About the Embedded PL/SQL Gateway
- About Oracle HTTP Server and mod_plsql
- About Password Security

About the Embedded PL/SQL Gateway

The embedded PL/SQL gateway installs with Oracle Database 11*g*. It provides the Oracle database with a Web server and also the necessary infrastructure to create dynamic applications. The embedded PL/SQL gateway runs in the Oracle XML DB HTTP server in the Oracle database and includes the core features of `mod_plsql`.

> **See Also:** "Configuring the Embedded PL/SQL Gateway" on page 4-15

About Oracle HTTP Server and mod_plsql

Oracle HTTP Server uses the `mod_plsql` plug-in to communicate to the Oracle Application Express engine within the Oracle database. It functions as communication broker between the Web server and the Oracle Application Express objects in the Oracle database. More specifically, it maps browser requests into database stored procedure calls over a `SQL*Net` connection.

> **See Also:** ""Configuring Oracle HTTP Server in a New Installation" on page 4-18

Note that this configuration consists of three tier architecture: a Web browser, Oracle HTTP Server (Apache) with `mod_plsql`, and an Oracle database containing Oracle Application Express.

About Password Security

If SSL is not used, then passwords could potentially be exposed, compromising the security of your Oracle Application Express instance.

Refer to "Configuring Secure Sockets Layer" on page 11 for more information.

Configuring the Embedded PL/SQL Gateway

Although the embedded PL/SQL gateway installs with the Oracle database, you must configure it before you can use it with Oracle Application Express. To accomplish, you run a configuration file and unlock the `ANONYMOUS` account.

Topics in this section include:

- Configuring the Embedded PL/SQL Gateway in New Installation or When Upgrading Database
- Disabling and Enabling the Oracle XML DB HTTP Server

> **See Also:** Choosing an HTTP Server on page 4-14 and About the Embedded PL/SQL Gateway on page 4-14

Configuring the Embedded PL/SQL Gateway in New Installation or When Upgrading Database

This section describes how to configure the embedded PL/SQL gateway by running the configuration script `apxconf.sql`. Running this script enables you to configure the port for Oracle XML DB HTTP server and the specify a password for the Oracle Application Express `ADMIN` account. Then, you unlock the `ANONYMOUS` account.

To configure the embedded PL/SQL gateway:

1. Change your working directory to `$ORACLE_HOME/apex`.
2. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS`. For example:

   ```
   $ $ORACLE_HOME/bin/sqlplus
   SQL> CONNECT SYS as SYSDBA
   Enter password: SYS_password
   ```

3. Run `apxconf.sql` as shown in the following example:

   ```
   @apxconf
   ```

4. When prompted, enter a password for the Oracle Application Express Admin account.

   Be sure to make a note of the password you enter. You will use this password to log in to Oracle Application Express Administration Services.

5. When prompted, enter the port for the Oracle XML DB HTTP server. The default port number is 8080.
6. Enter the following statement to unlock the ANONYMOUS account:

   ```
   ALTER USER ANONYMOUS ACCOUNT UNLOCK;
   ```

Disabling and Enabling the Oracle XML DB HTTP Server

The embedded PL/SQL gateway runs in the Oracle XML DB HTTP server in the Oracle database. This section describes how to enable or disable the Oracle XML DB HTTP server.

Topics in this section include:

- Disabling Oracle XML DB HTTP Server
- Enabling Oracle XML DB HTTP Server

See Also: Configuring the Embedded PL/SQL Gateway on page 4-15

Disabling Oracle XML DB HTTP Server

To disable Oracle XML DB HTTP server:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS`. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following statement:

```
EXEC DBMS_XDB.SETHTTPPORT(0);
COMMIT;
```

Enabling Oracle XML DB HTTP Server

To enable Oracle XML DB HTTP server:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS`. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following statement:

```
EXEC DBMS_XDB.SETHTTPPORT(port);
COMMIT;
```

   For example:

```
EXEC DBMS_XDB.SETHTTPPORT(8080);
COMMIT;
```

Note: Port numbers less than 1024 are reserved for use by privileged processes on many operating systems. To enable the XML DB HTTP listener on a port less than 1024, such as 80, review the following documentation:

- Refer to chapter "Using Protocols to Access the Repository" in *Oracle XML DB Developer's Guide* for more information on using HTTP(S) and Oracle XML DB Protocol Server.
- Refer to chapter "Protocol Address Configuration" in *Oracle Database Net Services Reference* for more information on protocol addressing.

Copying the Images Directory

Whether you are loading a new installation or upgrading from a previous release, you must copy the images directory from the top level of the `$ORACLE_HOME/apex` directory to the location on the file system containing the Oracle home for Oracle HTTP Server.

Note: This section is relevant only if you choose Oracle HTTP Server with mod_plsql. However, if you choose Oracle XML DB HTTP Server with the embedded PL/SQL gateway, then these steps can be ignored.

Topics in this section include:

- Copying the Images Directory After an Upgrade
- Copying the Images Directory After a New Installation

Copying the Images Directory After an Upgrade

During an upgrade, you must overwrite your existing images directory. Before you begin the upgrade, to ensure that you can revert to the previous version, Oracle recommends that you create a copy of your existing `images` directory for Oracle Application Express, indicating the release number of the images (for example, `images_2_0`).

To locate the `images` directory on the file system, review the following files for the text alias `/i/`:

- Oracle9*i* HTTP Server release 2, see the `httpd.conf` file
- Oracle HTTP Server distributed with Oracle Database 11*g*, see the `dads.conf` file
- Oracle Application Server 10*g*, see the `marvel.conf` file

When you locate the images directory path, Oracle recommends that you copy the existing images directory to a backup location. Doing this enables to revert to the previous release, if that becomes necessary.

After you copy the existing images directory, use the following command syntax to copy the `apex/images` directory from the 11*g* Oracle database home to the existing images directory path, overwriting the existing images:

- Oracle Application Server 10g:

```
cp -rf $ORACLE_HOME/apex/images ORACLE_HTTPSERVER_HOME/Apache
```

- Oracle HTTP Server distributed with Oracle Database 11*g*:

```
cp -rf $ORACLE_HOME/apex/images ORACLE_HTTPSERVER_HOME/ohs
```

In the preceding syntax example:

- ORACLE_HOME is the Oracle Database 11*g* Oracle home
- ORACLE_HTTPSERVER_HOME is the existing Oracle Application Server or Oracle HTTP Server Oracle home, such as /u01/app/oracle/db_2/

Copying the Images Directory After a New Installation

You can copy the apex/images directory by executing a command similar to the one shown in the following example:

```
cp -rf $ORACLE_HOME/apex/images ORACLE_HTTPSERVER_HOME/ohs
```

In the preceding syntax example:

- $ORACLE_HOME is the Oracle Database 11*g* Oracle home.
- ORACLE_HTTPSERVER_HOME is the existing Oracle Application Server or Oracle HTTP Server Oracle home, such as /u01/app/oracle/db_2/.

Configuring Oracle HTTP Server in a New Installation

This section describes how to configure Oracle HTTP Server with mod_plsql in a new installation.

Topics in this section include:

- Configuring Oracle HTTP Server Release 9.0.3 in a New Installation
- Configuring Oracle HTTP Server distributed with Oracle Database 11g or Oracle Application Server 10g in a New Installation

Configuring Oracle HTTP Server Release 9.0.3 in a New Installation

In Oracle HTTP Server release 9.0.3, the wdbsvr.app file contains information about the DAD to access Oracle Application Express. A DAD is a set of values that specify how the Oracle HTTP Server component modplsql connects to the database server to fulfill an HTTP request.

Topics in this section include:

- Changing the Password for the ADMIN Account
- Changing the Password for the APEX_PUBLIC_USER Database User
- Modifying the wdbsvr.app File in a New Installation
- Modifying the Oracle9i httpd.conf

Changing the Password for the ADMIN Account

To change the password for the ADMIN account:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run `apxxepwd.sql`. For example:

```
@apxxepwd.sql
```

When prompted enter a password for the ADMIN account.

Changing the Password for the APEX_PUBLIC_USER Database User

In order to specify the password in the `DAD` file, you have to change the password for the database user `APEX_PUBLIC_USER`. Please use the following steps to change the password for the `APEX_PUBLIC_USER` database user:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS`. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following command:

```
SQL>PASSWORD APEX_PUBLIC_USER
Changing password for APEX_PUBLIC_USER
New password: password
Retype new password: password
```

Modifying the wdbsvr.app File in a New Installation

To create the DAD, you modify the file `wdbsvr.app` and add an entry for Oracle Application Express.

To modify the `wdbsvr.app` file, follow these steps:

1. Using a text editor, open the following file:

```
ORACLE_HTTPSERVER_HOME/Apache/modplsql/cfg/wdbsvr.app
```

2. Add an entry for Oracle Application Express using the following syntax. Only change the settings indicated in italics.

```
[DAD_htmldb]
 connect_string = localhost:1521:orcl
 password = apex
 username = apex_public_user
 default_page = apex
 document_table = wwv_flow_file_objects$
 document_path = docs
 document_proc = wwv_flow_file_mgr.process_download
 reuse = Yes
 enablesso = No
 stateful = STATELESS_RESET
 nls_lang = American_America.AL32UTF8
```

Where:

- `connect_string` refers to the host ID, port number, and Oracle9*i* database where Oracle Application Express was installed. Use the format `host:port:sid`.

  If the Oracle9*i* version of Oracle HTTP Server you want to use is installed in the same Oracle home as the database you specified for use with Oracle Application Express, leave this parameter blank.

- `password` is the Oracle Application Express password for the Oracle Application ADMIN account you specified in Changing the Password for the APEX_PUBLIC_USER Database User on page 4-22.
- `nls_lang` determines the language setting of the DAD. The character set portion of the `nls_lang` value must always be set to `AL32UTF8`, regardless of whether or not the database character set is `AL32UTF8`.

  If either the territory portion or the language portion of the NLS settings contains a space, you must wrap the value in double quotes as shown in the following example:

  ```
  nls_lang = "ENGLISH_UNITED KINGDOM.AL32UTF8"
  ```

  You can find information about the database character set by querying the view `NLS_DATABASE_PARAMETERS` as shown in the following example:

  ```
  SELECT value
  FROM nls_database_parameters
  WHERE PARAMETER = 'NLS_CHARACTERSET';
  ```

3. Leave the remaining settings, including the username setting, as they appear in the previous example.
4. Save and exit the `wdbsvr.app` file.

Modifying the Oracle9*i* httpd.conf

You need to modify the `httpd.conf` file to include an alias that points to the file system path where you copied the images directory. You may also need to modify the `httpd.conf` file to add two new MIME types to support SQL Workshop.

> **See Also:** Copying the Images Directory After an Upgrade on page 4-17

To modify `httpd.conf` file, follow these steps:

1. Using a text editor, open the following file:

   ORACLE_HTTPSERVER_HOME/Apache/conf/httpd.conf

2. Add an alias entry that points to the file system path where you copied the images directory. The example is as follows:

   ```
   Alias /i/ "/home/oracle/OraHome1/Apache/Apache/images/"
   ```

3. Next, add two new MIME types to support SQL Workshop:

   Add the following lines if it does not currently exist:

   ```
   AddType text/xml            xbl
   AddType text/x-component    htc
   ```

   If you are upgrading from Oracle HTML DB 2.0, these MIME types should already exist.

4. Save and exit the `httpd.conf` file.
5. To stop and restart the Oracle HTTP Server, run the following command.

   ```
   ORACLE_HTTPSERVER_HOME/Apache/bin/apachectl stop
   ORACLE_HTTPSERVER_HOME/Apache/bin/apachectl start
   ```

Note that if the Oracle HTTP Server is listening on a port less than 1024, then these commands must be executed as a privileged user (such as `root`).

See Also: *Oracle HTTP Server Administrator's Guide*

Configuring Oracle HTTP Server distributed with Oracle Database 11*g* or Oracle Application Server 10*g* in a New Installation

Oracle Application Express must have access to Oracle HTTP Server with `mod_plsql`. Perform the following postinstallation steps if:

- This is a new installation of Oracle Application Express (that is, you are not upgrading from a previous release).
- You run Oracle HTTP Server distributed with Oracle Database 11*g* or Oracle Application Server 10*g*.
- Oracle HTTP Server is installed in an Oracle home.
- You have not previously configured Oracle HTTP Server to work with Oracle Application Express.

These instructions do not apply if you run Oracle HTTP Server Release 9.0.3. For more information on configuring Oracle HTTP Server Release 9.0.3, see "Configuring Oracle HTTP Server Release 9.0.3 in a New Installation" on page 4-18.

Topics in this section include:

- Changing the Password for the ADMIN Account
- Unlocking the APEX_PUBLIC_USER Database User
- Changing the Password for the APEX_PUBLIC_USER Database User
- Edit `dads.conf` File
- Stop and Restart Oracle HTTP Server

Note: The Oracle home directory `(ORACLE_HTTPSERVER_HOME)` is the location where Oracle HTTP Server is installed.

Changing the Password for the ADMIN Account

First, change the password for the Oracle Application Express `ADMIN` account.

To change the password for the `ADMIN` account:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as `SYS`. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run `apxxepwd.sql`. For example:

```
@apxxepwd.sql
```

When prompted enter a password for the ADMIN account.

Unlocking the APEX_PUBLIC_USER Database User

When configuring Oracle HTTP Server for Oracle Application Express in a new installation, the database user `APEX_PUBLIC_USER` must be an unlocked account. To

unlock the account for database user APEX_PUBLIC_USER, execute the following steps:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following command:

```
SQL> ALTER USER APEX_PUBLIC_USER ACCOUNT UNLOCK
```

Changing the Password for the APEX_PUBLIC_USER Database User

In order to specify the password in the DAD file, you have to change the password for the database user APEX_PUBLIC_USER. Please use the following steps to change the password for the APEX_PUBLIC_USER database user:

1. Start SQL*Plus and connect to the database where Oracle Application Express is installed as SYS. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Run the following command:

```
SQL>PASSWORD APEX_PUBLIC_USER
Changing password for APEX_PUBLIC_USER
New password: password
Retype new password: password
```

Edit dads.conf File

If this is a new installation of Oracle Application Express, then you must edit the dads.conf file. The dads.conf file contains the information about the DAD to access Oracle Application Express.

To edit the dads.conf file, follow these steps:

1. Using a text editor, edit the following file:
   - Oracle Application Server 10*g*:

     ORACLE_HTTPSERVER_HOME/Apache/modplsql/conf/dads.conf

   - Oracle HTTP Server distributed with Oracle Database 11*g*:

     ORACLE_HTTPSERVER_HOME/ohs/modplsql/conf/dads.conf

2. Copy the following into the dads.conf file. Replace ORACLE_HTTPSERVER_HOME, host, port, service_name, and apex_public_user_password with values appropriate for the environment. Note that apex_public_user_password is the same password you defined in Changing the Password for the APEX_PUBLIC_USER Database User on page 4-19.

   Note that the path listed is only an example. The path in the dads.conf file should reference the file system path described in Copying the Images Directory on page 4-17.

```
Alias /i/ "ORACLE_HTTPSERVER_HOME/images/"
AddType text/xml       xbl
```

```
AddType text/x-component       htc

<Location /pls/htmldb>
 Order deny,allow
 PlsqlDocumentPath docs
 AllowOverride None
 PlsqlDocumentProcedure      wwv_flow_file_manager.process_download
 PlsqlDatabaseConnectString  host:port:service_name ServiceNameFormat
 PlsqlNLSLanguage            AMERICAN_AMERICA.AL32UTF8
 PlsqlAuthenticationMode     Basic
 SetHandler                  pls_handler
 PlsqlDocumentTablename      wwv_flow_file_objects$
 PlsqlDatabaseUsername       APEX_PUBLIC_USER
 PlsqlDefaultPage            apex
 PlsqlDatabasePassword       apex_public_user_password
 Allow from all
</Location>
```

3. Locate the line containing `PlsqlNLSLanguage`.

   The `PlsqlNLSLanguage` setting determines the language setting of the DAD. The character set portion of the `PlsqlNLSLanguage` value must be set to `AL32UTF8`, regardless of whether or not the database character set is `AL32UTF8`. For example:

   ```
   PlsqlNLSLanguage            AMERICAN_AMERICA.AL32UTF8
   ```

4. Save and exit the `dads.conf` file.

Stop and Restart Oracle HTTP Server

To stop and restart Oracle HTTP Server,run the following commands:

```
ORACLE_HTTPSERVER_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
ORACLE_HTTPSERVER_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server
```

Enabling Network Services in Oracle Database 11*g*

By default, the ability to interact with network services is disabled in Oracle Database 11*g* Release 1 (11.1). Therefore, if you run Oracle Application Express with Oracle Database 11*g* Release 1 (11.1), you need to use the new `DBMS_NETWORK_ACL_ADMIN` package to grant connect privilege to any host for the `FLOWS_030000` database user. Failing to grant these privileges results in issues with the following:

- Sending outbound mail in Oracle Application Express.

  Users can call methods from the `APEX_MAIL` package, but issues arise when sending outbound email.

- Using Web services in Oracle Application Express.
- PDF/report printing.
- Searching for content in online Help (that is, using the Find link).

This section contains the following topics:

- Granting Connect Privileges
- Troubleshooting an Invalid ACL Error

Granting Connect Privileges

The following example demonstrates how to grant connect privileges to any host for the FLOWS_030000 database user.

In order to run the examples, the compatible initialization parameter of the database must be set to at least 11.1.0.0.0. In a 11*g* database, the parameter is already set by default. However, you will have to set this parameter in case of a database upgrade to 11*g* from a prior version.

> **See Also:** "Creating and Configuring an Oracle Database" in the *Oracle Database Administrator's Guide* for information about changing database compatible initialization parameters.

```
DECLARE
ACL_PATH VARCHAR2(4000);
ACL_ID RAW(16);
BEGIN
-- Look for the ACL currently assigned to '*' and give FLOWS_030000
-- the "connect" privilege if FLOWS_030000 does not have the privilege yet.
SELECT ACL INTO ACL_PATH FROM DBA_NETWORK_ACLS
WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

-- Before checking the privilege, make sure that the ACL is valid
-- (for example, does not contain stale references to dropped users).
-- If it does, the following exception will be raised:
--
-- ORA-44416: Invalid ACL: Unresolved principal 'FLOWS_030000'
-- ORA-06512: at "XDB.DBMS_XDBZ", line ...
--
SELECT SYS_OP_R2O(extractValue(P.RES, '/Resource/XMLRef')) INTO ACL_ID
FROM XDB.XDB$ACL A, PATH_VIEW P
WHERE extractValue(P.RES, '/Resource/XMLRef') = REF(A) AND
EQUALS_PATH(P.RES, ACL_PATH) = 1;

DBMS_XDBZ.ValidateACL(ACL_ID);

IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE(ACL_PATH, 'FLOWS_030000', 'connect')
IS NULL THEN
DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE(ACL_PATH,
'FLOWS_030000', TRUE, 'connect');
END IF;

EXCEPTION
-- When no ACL has been assigned to '*'.
WHEN NO_DATA_FOUND THEN
DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('power_users.xml',
'ACL that lets power users to connect to everywhere',
'FLOWS_030000', TRUE, 'connect');
DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL('power_users.xml','*');
END;
/
COMMIT;
```

Troubleshooting an Invalid ACL Error

If you receive an ORA-44416: Invalid ACL error after running the previous script, use the following query to identify the invalid ACL:

```
REM Show the dangling references to dropped users in the ACL that is assigned
REM to '*'.
```

```
SELECT ACL, PRINCIPAL
FROM DBA_NETWORK_ACLS NACL, XDS_ACE ACE
WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL AND
NACL.ACLID = ACE.ACLID AND
NOT EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);
```

Next, run the following code to fix ACL:

```
DECLARE
  ACL_ID   RAW(16);
  CNT      NUMBER;
BEGIN

-- LOOK FOR THE OBJECT ID OF THE ACL CURRENTLY ASSIGNED TO '*'
SELECT ACLID INTO ACL_ID FROM DBA_NETWORK_ACLS
WHERE HOST = '*' AND LOWER_PORT IS NULL AND UPPER_PORT IS NULL;

-- IF JUST SOME USERS REFERENCED IN THE ACL ARE INVALID, REMOVE JUST THOSE
-- USERS IN THE ACL. OTHERWISE, DROP THE ACL COMPLETELY.
SELECT COUNT(PRINCIPAL) INTO CNT FROM XDS_ACE
WHERE ACLID = ACL_ID AND
EXISTS (SELECT NULL FROM ALL_USERS WHERE USERNAME = PRINCIPAL);

  IF (CNT > 0) THEN

    FOR R IN (SELECT PRINCIPAL FROM XDS_ACE
                WHERE ACLID = ACL_ID AND
                      NOT EXISTS (SELECT NULL FROM ALL_USERS
                                    WHERE USERNAME = PRINCIPAL)) LOOP
      UPDATE XDB.XDB$ACL
         SET OBJECT_VALUE =
               DELETEXML(OBJECT_VALUE,
                         '/ACL/ACE[PRINCIPAL="'||R.PRINCIPAL||'"]')
       WHERE OBJECT_ID = ACL_ID;
    END LOOP;

  ELSE
    DELETE FROM XDB.XDB$ACL WHERE OBJECT_ID = ACL_ID;
  END IF;

END;
/

REM Commit the changes.

COMMIT;
```

Once the ACL has been fixed, you need to run the first script in this section to apply the ACL to the `FLOWS_030000` user. See "Granting Connect Privileges" on page 4-24.

Running Oracle Application Express in Other Languages

The Oracle Application Express interface is translated into German, Spanish, French, Italian, Japanese, Korean, Brazilian Portuguese, Simplified Chinese, and Traditional Chinese. A single instance of Oracle Application Express can be installed with one or more of these translated versions. At runtime, each user's Web browser language settings determine the specific language version.

The translated version of Oracle Application Express should be loaded into a database that has a character set that can support the specific language. If you attempt to install a translated version of Oracle Application Express into a database that does support the character encoding of the language, the installation may fail or the translated Oracle Application Express instance may appear corrupt when run. The database character set AL32UTF8 supports all the translated versions of Oracle Application Express.

You can manually install translated versions of Oracle Application Express using SQL*Plus. The installation files are encoded in AL32UTF8.

> **Note:** Regardless of the target database character set, to install a translated version of Oracle Application Express, you must set the character set value of the NLS_LANG environment variable to AL32UTF8 before starting SQL*Plus.

The following examples illustrate valid NLS_LANG settings for loading Oracle Application Express translations:

```
American_America.AL32UTF8
Japanese_Japan.AL32UTF8
```

Installing a Translated Version of Oracle Application Express

Whether you are installing for the first time or upgrading from a previous release, you must run the load_lang.sql script to run a translated version of Oracle Application Express.

The installation scripts are located in subdirectories identified by a language code in the unzipped distribution apex/builder. For example, the German version is located in /apex/builder/de and the Japanese version is located in /apex/builder/ja. Within each of these directories, there is a language loading script identified by the language code (for example, load_de.sql or load_ja.sql).

To install a translated version of Oracle Application Express:

1. Set the NLS_LANG environment variable, making sure that the character set is AL32UTF8. For example:
   - Bourne or Korn shell:

     ```
     $ NLS_LANG=American_America.AL32UTF8
     $ export NLS_LANG
     ```

   - C shell:

     ```
     % setenv NLS_LANG American_America.AL32UTF8
     ```

2. Start SQL*Plus and connect to the target database as SYS.

   ```
   $ $ORACLE_HOME/bin/sqlplus
   SQL> CONNECT SYS as SYSDBA
   Enter password: SYS_password
   ```

3. Execute the following statement:

   ```
   SQL> ALTER SESSION SET CURRENT_SCHEMA = FLOWS_030000;
   ```

4. Execute the appropriate language specific script. For example:

```
SQL> @load_de.sql
```

Where lang is the specific language (for example, `load_de.sql` for German or `load_ja.sql` for Japanese).

Managing JOB_QUEUE_PROCESSES

JOB_QUEUE_PROCESSES determine the maximum number of concurrently running jobs. In Oracle Application Express release 3.0, transactional support and SQL scripts require jobs. If JOB_QUEUE_PROCESSES is not enabled and working properly, you cannot successfully execute a script.

Topics in this section include:

- Viewing the Number of JOB_QUEUE_PROCESSES
- Changing the Number of JOB_QUEUE_PROCESSES

Viewing the Number of JOB_QUEUE_PROCESSES

There are currently three ways to view the number of JOB_QUEUE_PROCESSES:

- In the installation log file
- On the About Application Express page in Oracle Application Express
- From SQL*Plus

Viewing JOB_QUEUE_PROCESSES in the Installation Log File

After installing or upgrading Oracle Application Express to release 3.0, you can view the number of JOB_QUEUE_PROCESSES in the installation log files.

Viewing JOB_QUEUE_PROCESSES in Oracle Application Express

You can also view the number of JOB_QUEUE_PROCESSES on the About Oracle Application Express page.

To view the About Oracle Application Express page:

1. Log in to Oracle Application Express.
2. On the Administration list, click **About Application Express**.

   The current number JOB_QUEUE_PROCESSES displays at the bottom of the page.

Viewing JOB_QUEUE_PROCESSES from SQL*Plus

You can also view the number of JOB_QUEUE_PROCESSES from SQL*Plus by running the following SQL statement:

```
sql> SELECT VALUE FROM v$parameter WHERE NAME = 'job_queue_processes'
```

Changing the Number of JOB_QUEUE_PROCESSES

You can change the number of JOB_QUEUE_PROCESSES by running a SQL statement in SQL*Plus:

To update the number of JOB_QUEUE_PROCESSES:

1. Log in to the database as SYSDBA using SQL*Plus.
2. In SQL*Plus run the following SQL statement:

   ```
   SQL> ALTER SYSTEM SET JOB_QUEUE_PROCESSES = number
   ```

For example, running the statement `ALTER SYSTEM SET JOB_QUEUE_PROCESSES = 20` sets `JOB_QUEUE_PROCESSES` to 20.

Obfuscating PlsqlDatabasePassword Parameter

The `PlsqlDatabasePassword` parameter specifies the password for logging in to the database. You can use the `dadTool.pl` utility to obfuscate passwords in the `dads.conf` file.

You can find the `dadTool.pl` utility in the following directory:

ORACLE_HTTPSERVER_HOME`/ohs/modplsql/conf`

Obfuscating Passwords

To obfuscate passwords, run `dadTool.pl` by following the instructions in the `dadTool.README` file.

Logging In to Oracle Application Express

You access the Oracle Application Express home page in a Web browser. To view or develop Oracle Application Express applications, the Web browser must support JavaScript and the HTML 4.0 and CSS 1.0 standards. See "Browser Requirements" on page 2-12.

Topics in this section include:

- Oracle Application Express User Privileges
- Setting Up Your Local Environment

Oracle Application Express User Privileges

In the Oracle Application Express development environment, users log in to a shared work area called a **workspace**. Users are divided into four primary privileges:

- **Workspace administrators** are users who perform administrator tasks specific to a workspace such as managing user accounts, monitoring workspace activity, and viewing log files.
- **Developers** are users who create and edit applications. Developers can have their own workspace or share a workspace.
- **End users** have no development privileges. You define end users so that they can access applications that do not use an external authentication scheme.
- Oracle Application Express **administrators** are superusers that manage an entire hosted instance using the Oracle Application Express Administration Services application.

Setting Up Your Local Environment

How you set up Oracle Application Express depends upon your user privilege. If you are a **developer** accessing a hosted development environment, then an administrator must grant you access to a workspace. If you are an Oracle Application Express **administrator**, you must perform the following steps:

1. **Log in to** Oracle Application Express **Administration Services.** Oracle Application Express Administration Services is a separate application for managing an entire Oracle Application Express instance. You log in using the `ADMIN` account and password created or reset during the installation process.

2. **Specify a provisioning mode.** In Oracle Application Express Administration Services, you need to determine how the process of creating (or provisioning) a workspace will work in your development environment.
3. **Create a Workspace.** A **workspace** is a virtual private database allowing multiple users to work within the same Oracle Application Express installation while keeping their objects, data and applications private. Each workspace has a unique ID and name. An Oracle Application Express administrator can create a workspace manually or have users submit requests.
4. **Log in to a Workspace.** Once you create a workspace in Oracle Application Express Administration Services, return to the Oracle Application Express Login page and log in to that workspace.

> **See Also:** *Oracle Database 2 Day + Application Express Developer's Guide* or "Quick Start" in *Oracle Database Application Express User's Guide*

Patching Oracle Application Express 3.0

If you are already running Oracle Application Express 3.0, then check the Oracle Application Express page on the Oracle Technology Network (OTN) at (`http://www.oracle.com/technology/products/database/application_express/index.html`) URL for information about patch set releases or later versions of Oracle Application Express:

Upgrading to Oracle Database 11*g* will not patch an Oracle Application Express 3.0 instance to Oracle Application Express 3.0.1.

Postinstallation Database Configuration for Oracle Configuration Manager

If you have installed Oracle Configuration Manager in a home that contains a database, you must run a script to create a database account to collect database configuration collections. You must create this account in both **Connected** and **Disconnected** modes. Refer to "Oracle Configuration Manager" on page 1-11 for further information on these modes. The database account stores the PL/SQL procedures that collect the configuration information, and the account owns the database management system (DBMS) job that performs the collection. After the account has been set up, as login privileges are no longer required, the account is locked.

Note:

- Because the collected configuration data is not stored in the database, additional disk space is not required for the database.
- Because database configuration collections are performed using the database jobs, the `job_queue_process` initialization parameter must have a value greater than 0 for pre-10*g* databases only.

This section contains the following topics:

- Additional Step for Oracle Enterprise Manager Grid Control

Preparing Pre-9.2 Databases

Before running the `installCCRSQL.sh` script to prepare the database for configuration collection, you must perform the following steps for pre 9.2 databases:

1. Edit the `init`*`sid`*`.ora` file where `sid` is the database system identifier, and set the `UTL_FILE_DIR` parameter to include `$ORACLE_HOME/ccr/state` as one of the directories.

   If a server parameter file (`spfile`) is used, alter the `UTL_FILE_DIR` parameter using the following SQL*Plus command:

   ```
   SQL>alter system set utl_file_dir=value scope=spfile
   ```

   where `value` is equal to `$ORACLE_HOME/ccr/`state
2. Restart the database.

Instrumenting the Database for Configuration Collections

To configure the database for configuration collection, run the following script:

- Run the following command to create the `admin` directory

  ```
  $ORACLE_HOME/ccr/bin/setupCCR
  ```

- Run the following script, to configure the database for configuration collection:

  ```
  $ORACLE_HOME/ccr/admin/scripts/installCCRSQL.sh collectconfig -s SID -r
  SYSDBA-USER -p SYSDBA-PASSWORD
  ```

However, only to configure the database for configuration collection, run the following script:

```
$ORACLE_HOME/ccr/admin/scripts/installCCRSQL.sh collectconfig -s SID -r
SYSDBA-USER -p SYSDBA-PASSWORD
```

The `installCCRSQL.sh` script creates an Oracle Configuration Manager user and loads the PL/SQL procedure into the database defined by the `ORACLE_SID`. You can also specify the database *`SID`* by using the `-s` option in the command line as in the following example where the *`SID`* is `orcl`:

```
$ORACLE_HOME/ccr/admin/scripts/installCCRSQL.sh collectconfig -s orcl
```

By default, the connection to the database is through operating system authentication, "/as sysdba." To specify a different `SYSDBA` user and password, you can use these options:

-r *`SYSDBA-USER`*: The login name of the `SYSDBA` user

`-p SYSDBA-PASSWORD`: The password for the `SYSDBA` user

Note:

- If you specify the SYSDBA user without specifying the password, you will be prompted to enter the password.
- If you specify only the SYSDBA password without specifying the user name, the user SYS is used by default.
- If the Oracle Configuration Manager account already exists, when you run the installCCRSQL.sh script, it will be dropped and re-created.
- If you are upgrading from a 9.x database version to a 10.x version, you must run the installCCRSQL.sh script again to record the upgraded version.

Additional Step for E-Business Suites

If the database is used as a repository for an Oracle E-Business Suite, you must also run the following script from the ORACLE_HOME in which the E-Business database has been hosted:

```
$ORACLE_HOME/ccr/admin/scripts/installCCRSQL.sh ebs_collectconfig -u Oracle_
Applications_User
```

The -u parameter is mandatory. If you do not specify this parameter, the application prompts you for the Oracle Applications User. If the -u parameter is specified, you will be prompted for the Oracle Applications Password.

If you want to automate the install, you can run the installCCRSQL.sh script with an additional -w option to specify the Oracle Applications Password. For example:

```
$ORACLE_HOME/ccr/admin/scripts/installCCRSQL.sh ebs_
collectconfig -u Oracle_Applications_User -w Oracle_Applications_
Password
```

You can add the -s SID command to specify the SID of the Oracle Applications Database instance.

If you are not using operating system authentication to connect to the database, you must use the -r and -p parameters to specify the following:

-r SYSDBA-USER: The login name of the SYSDBA user

-p SYSDBA-PASSWORD: The password for the SYSDBA user

If the -r parameter is specified, the -p parameter is optional and will be prompted for.

Additional Step for Oracle Enterprise Manager Grid Control

If the database is used as a repository for Oracle Enterprise Manager Grid Control, you must also run the following script:

```
$ORACLE_HOME/ccr/admin/scripts/installCCRSQL.sh collectemrep
```

When you run this command, then the application prompts you for the SYSMAN password. If you want to automate the install, you can run the installCCRSQL.sh script to specify the SYSMAN password. For example:

```
$ORACLE_HOME/ccr/admin/scripts/installCCRSQL.sh collectemrep -e
SYSMAN PASSWORD
```

You can add the `-s SID` command to specify the `SID` of the Oracle Enterprise Manager Grid Control Database instance. You must run this script from the `ORACLE_HOME` in which the Oracle Enterprise Manager Grid Control database has been hosted.

If you are not using operating system authentication to connect to the database, you must use the `-r` and `-p` parameters to specify the following:

`-r SYSDBA-USER`: The login name of the `SYSDBA` user

`-p SYSDBA-PASSWORD`: The password for the `SYSDBA` user

If the `-r` parameter is specified, the `-p` parameter is optional and will be prompted for.

5

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
- Logging In to Oracle Enterprise Manager Database Control
- Managing Automatic Storage Management
- Accessing Oracle Database with SQL*Plus
- Accessing Oracle Database with SQL Developer
- 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, perform the following steps:

1. Start Oracle Universal Installer, follow the instructions in "Running Oracle Universal Installer" on page 3-8.
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 **Close** to close the Inventory dialog box.
6. Click **Cancel** to close Oracle Universal Installer, and then click **Yes** to confirm.

Logging In to Oracle Enterprise Manager Database Control

If you configured Oracle Enterprise Manager Database Control during the installation, you can use it to manage the database. Alternatively, you can use Oracle Enterprise Manager Grid Control to manage the database. To display the Database Control:

1. Use a Web browser to access the Database Control URL:

   http://*host*:*port*/em

   In this example:

   - *host* is the name of the computer on which you installed Oracle Database
   - *port* is the port number reserved for the Database Control during installation

   If you do not know the correct port number to use, look for the following line in the `$ORACLE_HOME/install/portlist.ini` file:

   ```
   Enterprise Manager Console HTTP Port (db_name) = 1158
   ```

   The installation reserves the first available port from the range 5500 to 5519. For example, if you installed Oracle Database on host `mgmt42`, and the Database Control uses port 1158, use the following URL:

   ```
   http://mgmt42:1158/em
   ```

   Oracle Enterprise Manager displays the Database Control login page.

2. Log in to the database using the user name `SYSTEM` and connect as `SYSDBA`. Enterprise Manager displays the Database Home page.

   Use the password that you specified for the `SYSTEM` account during the installation.

> **Note:** You can also log in to the Database Control using the `SYSTEM` or `SYSMAN` accounts or you can grant login privileges to other database users.

Understanding Database Control Login Privileges

When you log in to the Oracle Enterprise Manager Database Control using the `SYSMAN` user account, you are logging in as the Oracle Enterprise Manager super user. The `SYSMAN` account is automatically granted the privileges and privileges required to access all the management features provided by the Database Control.

You can also use the `SYS` and `SYSTEM` accounts to log in to the Database Control. In addition, you can grant login privileges to other database users, as follows:

1. Log in to the Database Control.

   > **See Also:** The "Logging In to Oracle Enterprise Manager Database Control" section for information about logging in to the Database Control

2. Click **Setup** at the top of the Database Control home page.
3. Click **Administrators** in the left navigation bar.
4. Click **Create** to create an Enterprise Manager user.
5. In the **Name** field, enter the user name of an existing database user or click the flashlight icon and select a user from the pop-up window.
6. Enter the password for this user, and then click **Review**.
7. On the properties page, click **Finish**.

Enterprise Manager assigns login privileges to the specified user and includes this user in the list of Enterprise Manager users on the Setup Administrators page.

Managing Automatic Storage Management

This section provides information about managing an Automatic Storage Management installation. It covers the following topics:

- Starting and Stopping Automatic Storage Management
- Automatic Storage Management Utilities

Starting and Stopping Automatic Storage Management

To start and stop Automatic Storage Management, refer to *Oracle Database Administrator's Reference for Linux and UNIX*.

Automatic Storage Management Utilities

To manage Automatic Storage Management, you can use the following tools:

- **asmcmd**: This command-line tool enables you to manage Automatic Storage Management disk group files and directories.
- **Oracle Enterprise Manager Grid Control**: If you have Oracle Enterprise Manager installed, you can use Grid Control to manage Automatic Storage Management functions, such as migrating an existing database to Automatic Storage Management, checking the status of the Automatic Storage Management instance, checking the performance of the Automatic Storage Management disk groups, and creating or dropping Automatic Storage Management disk groups.
- **Oracle Enterprise Manager Database Control**: This utility enables you to perform functions similar to Grid Control.
- **SQL*Plus**: You can run commands that are specific to Automatic Storage Management from either of these tools. To connect to an Automatic Storage Management instance, use the same methods that you use to connect to an Oracle database instance.

See Also:

- "Logging In to Oracle Enterprise Manager Database Control"
- *Oracle Database Administrator's Guide* for more information about managing Automatic Storage Management
- *Oracle Database Utilities* for more information about `asmcmd`

Accessing Oracle Database with SQL*Plus

To run the SQL and PL/SQL commands to access the Oracle Database, you can use SQL*Plus. This tool enables you to perform the same database management operations, and to query, insert, update, or delete data directly in the database.

Use the following command to start SQL*Plus and log in as the `SYS` user, connecting as `SYSDBA`:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

For example, to log on as `SYSTEM` using the password `Systempwd1`, you enter:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYSTEM
```

```
Enter password: Systempwd1
```

If you are logging on as `SYS`, you would need to connect as `SYSDBA`:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

See Also:

- *SQL*Plus User's Guide and Reference*
- *SQL*Plus Quick Reference*

Accessing Oracle Database with SQL Developer

To run the SQL and PL/SQL commands to access Oracle Database, you can use SQL Developer. All SQL and PL/SQL commands are supported as they are passed directly from the SQL Worksheet to the Oracle Database.

Set Up the JDK Path For SQL Developer

Set the following environmental variables to ensure that the correct jdk is picked up:

- `$ORACLE_HOME`
- `$JAVA_HOME=$ORACLE_HOME/jdk`
- `$PATH=$JAVA_HOME/bin/:$PATH`

To start SQL Developer on which the Sun Java SDK release 1.5 is installed, use the following commands:

- Change to `$ORACLE_HOME/sqldeveloper`.
- Run `$ ./sqldeveloper.sh`.
- Right-Click Connections. In the dialog box, enter a Connection name, username, password, and for the host string, the name of the database to which you want to connect and click Connect.

Once connected, you can view, create, modify, and delete the database objects using the Connection Navigator or issue any SQL or PL/SQL command using a SQL Worksheet (From the **Tools** menu, select **SQL Worksheet**).

SQL*Plus commands have to be interpreted by the SQL Worksheet before being passed to the database. The SQL Worksheet currently supports a number of SQL*Plus commands. SQL*Plus commands which are not supported by the SQL Worksheet are ignored and are not sent to the Oracle Database.

See Also:

"SQL*Plus Statements Supported and Not Supported in SQL Worksheet" in *Oracle Database SQL Developer User's Guide*

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 accounts, you must unlock them and reset their passwords. Table 5–1 describes these accounts and lists their user names and default passwords.

See Also: The "Unlocking and Resetting User Passwords" section for information about unlocking and resetting passwords.

Note: Use the Oracle Enterprise Manager Database Control to view the complete list of database accounts.

Table 5–1 Database Accounts

User Name	Description	See Also
ANONYMOUS	Allows HTTP access to Oracle XML DB.	None
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.	*Oracle Database Sample Schemas*
CTXSYS	The Oracle Text account.	*Oracle Text Reference*
DBSNMP	The account used by the Management Agent component of Oracle Enterprise Manager to monitor and manage the database. It is created only if you configure the database to use the Database Control.	*Oracle Enterprise Manager Grid Control Installation and Basic Configuration*
DIP	The account used by the Directory Integration Platform (DIP) to synchronize the changes in Oracle Internet Directory with the applications in the database.	None
EXFSYS	The account owns the Expression Filter schema.	None
FLOWS_030000	The account owns the Application Express schema and metadata.	*Oracle Database Application Express User's Guide*
FLOWS_FILES	The account owns the Application Express uploaded files.	*Oracle Database Application Express User's Guide*
APEX_PUBLIC_USER	The minimally privileged account used for Application Express configuration with Oracle HTTP Server and mod_plsql.	*Oracle Database Application Express User's Guide*
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.	*Oracle Database Sample Schemas*
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.	*Oracle Database Sample Schemas*
LBACSYS	The Oracle Label Security administrator account.	*Oracle Label Security Administrator's Guide*
MDDATA	The schema used by Oracle Spatial for storing Geocoder and router data.	*Oracle Spatial Developer's Guide*
MDSYS	The Oracle Spatial and Oracle Multimedia Locator administrator account.	*Oracle Spatial Developer's Guide*
MGMT_VIEW	An account used by Oracle Enterprise Manager Database Control.	None

Table 5–1 (Cont.) Database Accounts

User Name	Description	See Also
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.	*Oracle Database Sample Schemas*
ORDPLUGINS	The Oracle Multimedia user. Plugins supplied by Oracle and third-party plugins are installed in this schema.	*Oracle Multimedia Reference*
ORDSYS	The Oracle Multimedia administrator account.	*Oracle Multimedia Reference*
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 privilege to centrally manage metadata associated with stored outlines.	*Oracle Database Concepts*
ORACLE_OCM	This account contains the instrumentation for configuration collection used by the Oracle Configuration Manager.	*Oracle Configuration Manager Installation and Administration Guide*
OWBSYS	The account used by Oracle Warehouse Builder as its default repository. You must unlock this account subsequent to installing the Oracle Database and before launching the Warehouse Builder Repository Assistant.	*Oracle Warehouse Builder Installation and Administration Guide*
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.	*Oracle Database Sample Schemas*
SCOTT	An account used by Oracle sample programs and examples.	*Oracle Database Administrator's Guide*
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.	*Oracle Database Administrator's Guide*
SI_INFORMTN_SCHEMA	The account that stores the information views for the SQL/MM Still Image Standard.	*Oracle Multimedia Reference*
SYS	The account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
SYSMAN	The account used to perform Oracle Enterprise Manager database administration tasks.It is created only if you configure the database to use the Database Control.	*Oracle Enterprise Manager Grid Control Installation and Basic Configuration*
SYSTEM	Another account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
WMSYS	The account used to store the metadata information for Oracle Workspace Manager.	*Oracle Database Workspace Manager Developer's Guide*
WKPROXY	The Ultra Search proxy user.	*Oracle Ultra Search Administrator's Guide*

Table 5–1 (Cont.) Database Accounts

User Name	Description	See Also
WK_TEST	The default Ultra Search instance schema.	*Oracle Ultra Search Administrator's Guide*
WKSYS	The account used to store Ultra Search system dictionaries and PL/SQL packages.	*Oracle Ultra Search Administrator's Guide*
XDB	The account used for storing Oracle XML DB data and metadata.	*Oracle XML DB Developer's Guide*
DVSYS	There are two privileges assciated with this account. Database Vault owner privilege manages the Database Vault privileges and configurations. The Database Vault Account Manager is used to manage database user accounts. **Note**: Part of Oracle Database Vault user interface text is stored in database tables in the DVSYS schema. By default, only the English language is loaded into these tables. You can use Oracle Database Vault Configuration Assistant to add more languages to Oracle Database Vault. For the necessary steps, refer to Appendix C in *Oracle Database Vault Administrator's Guide*	*Oracle Database Vault Administrator's Guide*

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, you must unlock it, using one of the following methods:

- Using Database Control to Unlock Accounts and Reset Passwords
- Using SQL*Plus to Unlock Accounts and Reset Passwords
- Unlocking and Changing Passwords

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

Using Database Control to Unlock Accounts and Reset Passwords

To unlock and reset user account passwords using Oracle Enterprise Manager Database Control:

1. Log in to the Database Control.

   > **See Also:** The "Logging In to Oracle Enterprise Manager Database Control" section for information about logging in to the Database Control

2. Click **Server**.

3. In the Security section of the Server page, click **Users**.

   Enterprise Manager displays a table listing all database accounts. The Account Status column indicates whether the account is locked and whether the password is expired.

4. Select the user account that you want to modify, then click **Edit**.

5. Use the General page of the Users property sheet to unlock the account and, optionally, to change the password.

   See Also: Click **Help** in the Database Control window for more information about using the Database Control.

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`:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Enter a command similar to the following, where *`account`* is the user account that you want to unlock and *`password`* is the new password:

```
SQL> PASSWORD account UNLOCK;
Changing password for account
New password: password
Retype new password: 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 the 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

Unlocking and Changing Passwords

Passwords for all Oracle system administration accounts except `SYS`, `SYSTEM`, `SYSMAN`, and `DBSNMP` are revoked after installation. Before you use a locked account, you must unlock it and reset its password. If you created a starter database during the installation, Oracle Database Configuration Assistant displays a screen with your database information and the Password Management button. Use the Password Management button to unlock only the user names you will use.

Apply the following guidelines when specifying passwords:

- Passwords must be between 8 and 30 characters long.

- Passwords must be from the ASCII character set.
- Passwords must not start with a numeral.
- Passwords must not be the same as the user name.
- Passwords must not be Oracle reserved words.
- The SYS account password must not be change_on_install.
- The SYSTEM account password must not be manager.
- The SYSMAN account password must not be sysman.
- The DBSNMP account password must not be dbsnmp.
- If you choose to use the same password for all the accounts, then that password must not be change_on_install, manager, sysman, or dbsnmp.
- Passwords should have at least one alphabetic, one numeric, and one special character.
- Passwords should not be simple or obvious words, such as welcome, account, database, and user.
- Passwords should not have any consecutive repeating characters.

> **See Also:** "Reviewing Accounts and Passwords" for more information about accounts and passwords

Identifying Databases

The Oracle Database 11*g* 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.oracle.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.oracle.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 `spfile`*`sid`*`.ora`, which is stored in the `$ORACLE_HOME/dbs` directory. However, if you choose Automatic Storage Management for the database, Database Configuration Assistant typically uses the same storage mechanism for the server parameter file.

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

You can use the Oracle Enterprise Manager Database Control to view the location of the server parameter file and list all of the initialization parameters, as follows:

1. Log in to the Database Control.

   See Also: The "Logging In to Oracle Enterprise Manager Database Control" section for information about logging in to the Database Control

2. Click **Server**.
3. In the Database Configuration section of the Server page, click **Initialization Parameters**.

   Enterprise Manager displays a table listing the current value of each initialization parameter.
4. Select **SPFile** tab.

   Enterprise Manager displays a table listing the value of each initialization parameter specified in the server parameter file. The location of the server parameter file is displayed in the earlier table.

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 11*g* databases.

Table 5–2 describes the tablespaces provided by the default preconfigured database.

Table 5–2 Tablespaces and Data Files

Tablespace	Data File	Description
EXAMPLE	EXAMPLE01.DBF	Stores the Sample Schemas, if you included them.
SYSAUX	SYSAUX01.DBF	Serves as an auxiliary tablespace to the SYSTEM tablespace. Some products and options that previously used the SYSTEM tablespace now use the SYSAUX tablespace to reduce the load on the SYSTEM tablespace.
SYSTEM	SYSTEM01.DBF	Stores the data dictionary, including definitions of tables, views, and stored procedures needed by the Oracle Database. Information in this area is maintained automatically.
TEMP	TEMP01.DBF	Stores temporary tables and indexes created during the processing of your SQL statement. If you run a SQL statement that involves a lot of sorting, such as the constructs GROUP BY, ORDER BY, or DISTINCT, then you may need to expand this tablespace.
UNDOTBS	UNDOTBS01.DBF	Stores undo information. The **undo tablespace** contains one or more undo segments that maintain transaction history that is used to roll back, or undo, changes to the database. All starter databases are configured to run in automatic undo management mode.
USERS	USERS01.DBF	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

To use the Oracle Enterprise Manager Database Control to view the list of data files used by the database and their associated tablespaces:

1. Log in to the Database Control.

   > **See Also:** The "Logging In to Oracle Enterprise Manager Database Control" section for information about logging in to the Database Control

2. Click **Server**.
3. In the Storage section of the Server page, click **Datafiles**.

   Enterprise Manager displays a table listing each data file, and the tablespace with which it is associated.

   > **See Also:** For more information about using the Database Control to view, modify, and create tablespaces, click **Help** in the Database Control window.

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 11*g* 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 User's Guide* for more information about redo log files

To use the Oracle Enterprise Manager Database Control to view or modify the redo log files for the preconfigured database:

1. Log in to the Database Control.

   > **See Also:** The "Logging In to Oracle Enterprise Manager Database Control" section for information about logging in to the Database Control

2. Click **Server**.
3. In the Storage section of the Server page, click **Redo Log Groups**.

   Enterprise Manager displays a table listing the redo log groups used by the database.
4. To view the name and location of the redo log file associated with a particular group, select that group then click **View**.

   > **See Also:** For more information about using the Database Control to view, modify, and create redo log files, click **Help** in the Database Control window

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 11*g* 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.

To use the Oracle Enterprise Manager Database Control to view information about the control files for the preconfigured database:

1. Log in to the Database Control.

   > **See Also:** "Logging In to Oracle Enterprise Manager Database Control" for information about logging in to the Database Control

2. Click **Server**.
3. In the Storage section of the Server page, click **Control Files**.

   Enterprise Manager displays a table listing the control files used by the database.

See Also: For more information about using the Database Control to view information about control files and creating backups of these files to trace them, click **Help** in the Database Control window

For more information about setting the `CONTROL_FILES` initialization parameter value, refer to *Oracle Database Administrator's Guide*

6

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:

> **See Also:** If you want to remove an Oracle RAC installation, refer to *Oracle Clusterware Installation Guide for Linux* and *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for more information
>
> If you want to remove an individual product, refer to the product-specific documentation for requirements and restrictions

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 and Automatic Storage Management instances.
- Shut down processes.
- Reconfigure the Oracle Cluster Synchronization Services Daemon, if necessary.
- Remove the Oracle software.

Identifying All Instances

To identify all instances associated with the Oracle home that you want to remove, enter the following command:

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

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

```
+ASM:/u01/app/oracle/product/11.1.0/db_1:N
CUST:/u01/app/oracle/product/11.1.0/db_1:N
```

These entries show that the +ASM Automatic Storage Manager instance and the CUST Oracle database instance are associated with the /u01/app/oracle/product/11.1.0/db_1 Oracle home directory.

Removing Oracle Configuration Manager

To uninstall Oracle Configuration Manager, follow these steps:

1. If the $ORACLE_HOME directory contains a database, remove the Oracle Configuration Manager user and the associated objects from the database by running the following script:

   ```
   SQL> $ORACLE_HOME/ccr/admin/scripts/dropocm.sql
   ```

2. If the database is a repository for the Oracle E-Business Suite, log in to the database as an SYSDBA user and remove the additional objects from the database by running the following script:

   ```
   $ORACLE_HOME/ccr/admin/scripts/ebs_dropccr.sql Oracle_Applications_User
   ```

3. If the database is a repository for Oracle Grid Control, log in to the database as the SYSMAN user and remove the additional objects from the database by running the following script:

   ```
   $ORACLE_HOME/ccr/admin/scripts/dropemrep_collect.sql
   ```

4. To stop the Scheduler and remove the service or the crontab entry, enter the following command:

   ```
   $ORACLE_HOME/ccr/bin/deployPackages -d $ORACLE_HOME/ccr/inventory/core.jar
   ```

5. Delete the ccr directory by entering the following command:

   ```
   $ rm -rf $ORACLE_HOME/ccr
   ```

   Oracle Configuration Manager is successfully uninstalled.

Removing Oracle Application Express from the Database

This section describes how to remove the Oracle Application Express schema, synonyms, and users from the database without deleting the database. If you are going to delete the database, then you do not need to complete these steps.

After using Oracle Universal Installer to remove Oracle Application Express from its Oracle home, you can remove Oracle Application Express components from the database. Perform the following steps:

> **Note:** You should not follow these steps if you have upgraded your database from a prior release, and still want to use the prior release of Oracle Application Express.

1. Use SQL*Plus to connect to the database as the privileged user SYS, for example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

2. Execute the following commands:

```
SQL> ALTER SESSION SET CURRENT_SCHEMA = FLOWS_030000;
SQL> EXEC wwv_flow_upgrade.drop_public_synonyms;
SQL> ALTER SESSION SET CURRENT_SCHEMA = SYS;
SQL> DROP USER FLOWS_030000 CASCADE;
SQL> DROP USER flows_files CASCADE;
SQL> DROP USER apex_public_user CASCADE;
```

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. If you want to keep this data, make sure 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 that you want to remove, for example:
   - Bourne, Bash, or Korn shell:

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

   - C shell:

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

3. At the prompt, specify the `SID` for the database that you want to remove.
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 that you want to delete, 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. If you want to remove another database, click **Yes** and repeat steps 6 through 8.

Note: You cannot perform an Oracle database installation from the same Oracle Universal Installer session in which you perform a deinstallation of Oracle database. In other words, if you deinstall Oracle database with Oracle Universal Installer and want to perform another Oracle database installation. then you must start a new Oracle Universal Installer session.

Removing an Automatic Storage Management Instance

To completely remove Oracle database software, you must also remove any Automatic Storage Management instances running in the Oracle home. To remove an Automatic Storage Management instance:

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

   ```
   $ su - oracle
   ```

2. Run the `oraenv` or `coraenv` script to set the environment for the Automatic Storage Management instance that you want to remove, for example:

   - Bourne, Bash, or Korn shell:

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

   - C shell:

     ```
     $ source /usr/local/bin/coraenv
     ```

3. At the prompt, specify the `SID` for the Automatic Storage Management instance that you want to remove.

4. Connect to the Automatic Storage Management instance as a `SYS` user:

   ```
   $ $ORACLE_HOME/bin/sqlplus
   SQL> CONNECT SYS as SYSASM
   Enter password: SYS_password
   ```

5. Enter the following command to determine whether any Oracle database instance is using the Automatic Storage Management instance:

   ```
   SQL> SELECT INSTANCE_NAME FROM V$ASM_CLIENT;
   ```

   This command lists all of the database instances that are using this Automatic Storage Management instance.

   Note: This command only lists database instances that run. It is possible that other instances are associated with the Automatic Storage Management instance, but they are not currently running.

   If you removed a database from this Oracle home but the output from the command shows that this Automatic Storage Management instance is supporting a database instance in another Oracle home, do not remove the Automatic Storage Management instance or the Oracle home.

6. If there are no database instances associated with this Automatic Storage Management instance, drop the disk groups associated with this instance as follows:

> **Note:** Dropping the Automatic Storage Management disk group makes the disk device available for use with another Automatic Storage Management instance if required. However, all data in the disk group is lost. Make sure that no other database instance requires any data from this disk group before you drop it.

a. Identify the disk groups associated with the Automatic Storage Management instance:

```
SQL> SELECT NAME FROM V$ASM_DISKGROUP;
```

b. For each disk group that you want to delete, enter a command similar to the following:

```
SQL> DROP DISKGROUP name INCLUDING CONTENTS;
```

7. Enter the following command to shut down the Automatic Storage Management instance:

```
SQL> SHUTDOWN
```

8. Remove the entry for the Automatic Storage Management instance from the `/etc/oratab` file.

Reconfiguring Oracle Cluster Synchronization Services

Oracle Cluster Synchronization Services (CSS) is a daemon process that is configured by the `root.sh` script when you configure an Automatic Storage Management instance. It is configured to start every time the system boots. This daemon process is required to enable synchronization between Oracle Automatic Storage Management and database instances. It must be running if an Oracle database is using Automatic Storage Management for database file storage.

> **Note:** On cluster systems with Oracle RAC installations, the CSS daemon is configured during the Oracle Clusterware installation. If the system is running Oracle Clusterware, refer to *Oracle Real Application Clusters Installation Guide for Linux and UNIX*for information about removing Oracle RAC or Oracle Clusterware.

Before you remove an Oracle Database 11*g* Oracle home, you must determine whether the CSS daemon is running from that Oracle home and whether any other Oracle Database 11*g* Oracle homes exist on the system:

- If the Oracle Database 11*g* Oracle home that you want to remove is the only Oracle Database 11*g* installation on the system, you can delete the CSS daemon configuration.
- If the CSS daemon is running from the Oracle Database 11*g* Oracle home that you want to remove and other Oracle Database 11*g* installations exist on the system, you must reconfigure the CSS daemon to run from another Oracle Database 11*g* Oracle home.

The following sections describe how to complete these tasks:

- Identifying Oracle Database 11g Oracle Homes
- Reconfiguring the Oracle CSS Daemon

- Deleting the Oracle CSS Daemon Configuration

Identifying Oracle Database 11*g* Oracle Homes

To identify all of the Oracle Database 11*g* Oracle home directories, enter the following command:

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

From the output, identify any Oracle home directories where Oracle Database 11*g* is installed. Oracle homes that contain Oracle Database 11*g* typically have paths similar to the following. However, they might use different paths.

```
/mount_point/app/oracle/product/11.1.0/db_n
```

If there is only one Oracle home directory that contains Oracle Database 11*g*, refer to the "Deleting the Oracle CSS Daemon Configuration" section on page 6-7 for information about deleting the Oracle CSS daemon configuration.

If you identify more than one Oracle Database 11*g* Oracle home directory, refer to the following section for information about reconfiguring the Oracle CSS daemon.

Reconfiguring the Oracle CSS Daemon

To reconfigure the Oracle CSS daemon so that it runs from an Oracle home that you are not removing, follow these steps:

1. In all Oracle home directories on the system, stop all Oracle Automatic Storage Management instances and any Oracle Database instances that use Automatic Storage Management for database file storage.
2. Switch user to `root`.
3. Enter the following command to identify the Oracle home directory being used to run the CSS daemon:

   ```
   # more /etc/oracle/ocr.loc
   ```

   The output from this command is similar to the following:

   ```
   ocrconfig_loc=/u01/app/oracle/product/11.1.0/db_1/cdata/ \
   localhost/local.ocr
   local_only=TRUE
   ```

   The `ocrconfig_loc` parameter specifies the location of the Oracle Cluster Registry (OCR) used by the CSS daemon. The path up to the `cdata` directory is the Oracle home directory where the CSS daemon is running (`/u01/app/oracle/product/11.1.0/db_1` in this example).

   Note: If the value of the `local_only` parameter is `FALSE`, Oracle Clusterware is installed on this system.

   See Also: *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for information about removing Oracle Real Applications Clusters or Oracle Clusterware

   If this Oracle home directory is not the Oracle home that you want to remove, go to the "Removing Oracle Software" section on page 6-8.

4. Change directory to the Oracle home directory for an Oracle Database 11*g* installation that you are *not* removing.
5. Set the `ORACLE_HOME` environment variable to specify the path to this Oracle home directory:
   - Bourne, Bash, or Korn shell:

     ```
     # ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_2;
     # export ORACLE_HOME
     ```

   - C shell:

     ```
     # setenv ORACLE_HOME /u01/app/oracle/product/11.1.0/db_2
     ```

6. Enter the following command to reconfigure the CSS daemon to run from this Oracle home:

   ```
   # $ORACLE_HOME/bin/localconfig reset $ORACLE_HOME
   ```

   This command stops the Oracle CSS daemon, reconfigures it in the new Oracle home, and then restarts it. When the system boots, the CSS daemon starts automatically from the new Oracle home.
7. To remove the original Oracle home directory, refer to "Removing Oracle Software" section on page 6-8.

Deleting the Oracle CSS Daemon Configuration

To delete the Oracle CSS daemon configuration, follow these steps:

> **Note:** Delete the CSS daemon configuration only if you are certain that no other Oracle Database 11*g* installation requires it.

1. Remove any databases or Automatic Storage Management instances associated with this Oracle home. Refer to the preceding sections for information about how to complete these tasks.
2. Switch user to `root`.
3. Change directory to the Oracle home directory that you are removing.
4. Set the `ORACLE_HOME` environment variable to specify the path to this Oracle home directory:
   - Bourne, Bash, or Korn shell:

     ```
     # ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1
     # export ORACLE_HOME
     ```

   - C shell:

     ```
     # setenv ORACLE_HOME /u01/app/oracle/product/11.1.0/db_1
     ```

5. Enter the following command to delete the CSS daemon configuration from this Oracle home:

   ```
   # $ORACLE_HOME/bin/localconfig delete
   ```

   The script stops the Oracle CSS daemon, then deletes its configuration. When the system boots, the CSS daemon no longer starts.

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 that you want to remove:

   - Bourne, Bash, or Korn shell:

     ```
     $ ORACLE_HOME=/u01/app/oracle/product/11.1.0/db_1
     $ export ORACLE_HOME
     ```

   - C shell:

     ```
     % setenv ORACLE_HOME /u01/app/oracle/product/11.1.0/db_1
     ```

3. Remove any databases or Automatic Storage Management instances associated with this Oracle home and delete or reconfigure the Oracle CSS daemon.

   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
Database Control	`$ORACLE_HOME/bin/emctl stop dbconsole`
Oracle Net listener	`$ORACLE_HOME/bin/lsnrctl stop`

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 that you want to remove, then click **Remove**.

   > **Note:** If you choose to remove Oracle JVM, Oracle Universal Installer removes all installed products that depend on Oracle JVM, including Oracle Database 11*g*.

   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**.

A

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 Net Configuration Assistant 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 prompt. It includes information about the following topics:

- Reasons for Using Silent Mode or Noninteractive Mode
- Creating a Database Using Automatic Storage Management as the Storage Option for Database Files
- General Procedure for Using Response Files

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 response file's variable settings is to pass them as command line arguments when you run Oracle Universal Installer. For example:

```
-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 if you do not want to embed sensitive information, such as passwords, in the response file. For example:

```
-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 several reasons why you might want to run Oracle Universal Installer in silent mode or suppressed mode.

Mode	Uses
Silent	Use silent mode if you want to: ■ Complete an unattended installation, which you might schedule using operating system utilities such as `at` ■ Complete several similar installations on multiple systems without user interaction ■ Install the software on a system that does not have X Window System software installed on it 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.
Suppressed (noninteractive)	Use suppressed mode if you want to complete similar Oracle software installations on more than one system, providing default answers to some, but not all of Oracle Universal Installer prompts. 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.

Creating a Database Using Automatic Storage Management as the Storage Option for Database Files

Before you create a database that uses Automatic Storage Management, you must run the `root.sh` script. For this reason, you cannot create a database using Automatic Storage Management as the storage option for database files during a silent-mode installation. Instead, you can complete a software-only installation using silent-mode, and then run the Oracle Net Configuration Assistant and Database Configuration Assistant configuration assistants in silent mode after you have completed the software-only installation and you have run the `root.sh` script.

> **Note:** This limitation applies only to databases that use Automatic Storage Management as the storage option for database files. You can create a database that uses the file system option during a silent-mode installation.

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 Net Configuration Assistant 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 already exist. This file specifies the location of the Oracle Inventory directory where Oracle Universal Installer creates the inventory of Oracle products installed on the system.

> **Note:** If Oracle software has been installed previously on the system, the `oraInst.loc` file might already exist. If the file does exist, you do not need to create a file.

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

1. Switch user to `root`:

```
$ su - root
```

2. Create the `/etc/` directory if it does not exist:

```
# mkdir -p /var/opt/oracle
# mkdir /etc/
```

3. Change directory as follows:

```
# cd /etc/
```

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

```
inventory_loc=$ORACLE_BASE/oraInventory
inst_group=oinstall
```

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

5. 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 at `database/response` directory on the installation media.

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

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

Table A–1 Response Files

Response File	Description
`enterprise.rsp`	Enterprise Edition installation of Oracle Database 11*g*
`standard.rsp`	Standard Edition installation of Oracle Database 11*g*
`custom.rsp`	Custom installation of Oracle Database 11*g*
`dbca.rsp`	Database Configuration Assistant
`netca.rsp`	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 may want to 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 "Silent-Mode Response File Error Handling" section on page G-7 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
   ```

   Note: A fully specified response file for an Oracle Database installation contains the passwords for database administrative accounts and for a user who 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, create the `oraInst.loc` file, as described in the previous section.

3. Ensure that the Oracle software owner user (typically `oracle`) has permissions to create or write to the Oracle home path that you will specify when you run Oracle Universal Installer.

4. To record a response file, enter a command similar to the following to start Oracle Universal Installer:

   > **Note:** Do not specify a relative path to the response file. If you specify a relative path, 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 you want to record the responses that you enter in a response file
   - *`response_filename`* is the full path and file name of the response file that you want to record

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 the `runInstaller` command with the `-help` option, for example:

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

The help information appears in a window after some time.

To run Oracle Universal Installer 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, 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:

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

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

   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 you want to run Oracle Universal Installer in silent mode.
   - `-noconfig` suppresses running the configuration assistants during installation, and a software-only installation is performed instead.
   - *`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:

```
$ sudo sh
password:
# /oracle_home_path/root.sh
```

Running Net Configuration Assistant Using a Response File

You can run Net Configuration Assistant 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 Net Configuration Assistant in silent mode, you must copy and edit a response file template. Oracle provides a response file template named `netca.resp` 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 `database/response` directory.

To run Net Configuration Assistant 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, 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:** Net 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. Enter a command similar to the following to run Net Configuration Assistant in silent mode:

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

   In this command:

   - The `/silent` option indicates that you want to run Net Configuration Assistant 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 the 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

Use `-progressOnly` flag to set the mode to noninteractive. In the 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

Use `-silent` flag to set the mode to silent. In the silent mode, Database Configuration Assistant uses values that you specify, in the response file or as command line options, to create a 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, 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, 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} -responseFile \
/local_dir/dbca.rsp
```

In this example:

- The -silent option indicates that you want to run Database Configuration Assistant in silent mode.
- The -progressOnly option indicates that you want to run Database Configuration Assistant in noninteractive mode.
- *local_dir* is the full path of the directory where you copied the dbca.rsp response file template.

B

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 are performing multiple Oracle Database installations, then you may want to 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 you are cloning has had patches applied to it. When you clone this Oracle home, the new Oracle home will have the patch updates as well.

When cloning Database Oracle homes using 11.1 Database Control, you need to update the exclude file list. This file list specifies files that need not be included when the source Oracle home is archived because these files are not required for the clone operation. The following files should not be included in the archive:

- `sqlnet.ora`
- `tnsnames.ora`
- `listener.ora`
- `oratab`

> **Note:** In addition to cloning an Oracle home, you can clone individual Oracle Database installations by using Enterprise Manager Database Control. *Oracle Database Administrator's Guide* provides detailed information about cloning Oracle Database installations and Oracle homes.

To clone an Oracle home:

1. Verify that the installation of Oracle Database that you want to clone has been successful.

   You can do this by reviewing the `installActions`*`date_time`*`.log` file for the installation session, which is normally located in the `/`*`orainventory_location`*`/logs` directory.

   If you have 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 "Removing Oracle Software" section on page 6-8 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/11.1.0/db_1`, then you zip the `db_1` directory by using the following command:

   ```
   # zip -r db_1.zip /u01/app/oracle/product/11.1.0/db_1
   ```

   Leave out the `admin`, `flash_recovery_area`, and `oradata` directories that are in the `11.1.0` directory. These directories will be created in the target installation later, when you create a new database there.

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 you want to clone the Oracle home, 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_BASE="target_oracle_
   base" ORACLE_HOME="target_oracle_home" ORACLE_HOME_NAME="unique_name_on node"
   [-responseFile full_directory_path]
   ```

   For example:

   ```
   $ORACLE_HOME/oui/bin/runInstaller -silent -clone ORACLE_BASE=
   "/u01/app/oracle/" ORACLE_HOME="/u01/app/oracle/product/11.1.0/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 `cloneActions`*`timestamp`*`.log` file. This log file is normally located in `/`*`orainventory_location`*`/logs` directory.

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

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

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

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

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 databases, and cloning an Oracle Database home

Use the following steps to configure Oracle Configuration Manager for a cloned Oracle home:

1. Run the `emSnapshotEnv` script from `bin` directory as follows:

   ```
   $ORACLE_HOME/ccr/bin/emSnapshotEnv
   ```

2. Copy the content of the `core.jar` into `pending` directory as follows:

   ```
   cp ccr/inventory/core.jar $ORACLE_HOME/ccr/inventory/pending
   ```

3. Use the following command to remove the previous state files:

   ```
   rm ORACLE_HOME/ccr/state/*.ll
   ```

4. If you have removed the state files, then you must relink the core functions with the following command:

   ```
   $ORACLE_HOME/ccr/bin/deployPackages
   ```

5. Use the following command to rerun Oracle Configuration Manager:

   ```
   $ORACLE_HOME/ccr/bin/configCCR
   ```

C

Using NAS Devices

If you have a network attached storage (NAS) device that has been certified through the Oracle Storage Compatibility Program (OSCP), then you can use it to store the Oracle software, the Oracle database files, or both. This appendix provides guidelines for using a NAS storage device for Oracle software and database files. It includes information about the following:

- General Configuration Guidelines for NAS Devices
- NFS Feature Description
- Choosing Mount Points
- Creating Files on a NAS Device for Use with Automatic Storage Management
- NFS Mount Options

General Configuration Guidelines for NAS Devices

Refer to the documentation provided with the NAS device for specific information about how to configure it. In addition, use the following guidelines to ensure that the performance of the Oracle software meets the requirements:

- Before using the NAS device for the installation, verify that it is certified.

  Note: The OSCP certifies NAS devices only for single-instance databases. For Oracle RAC installations, refer to the Certify page on the Oracle*MetaLink* Web site for information about certified storage solutions, including NAS:

  `https://metalink.oracle.com`

- The performance of Oracle software and databases stored on NAS devices depends on the performance of the network connection between the Oracle server and the NAS device.

  For this reason, Oracle recommends that you connect the server to the NAS device using a private dedicated network connection, which should be Gigabit Ethernet or better.

- For single instance installations, you must create a separate Oracle home directory for each installation. Run the software in this Oracle home directory only from the system that you used to install it.

NFS Feature Description

The following are the features of NFS:

- Oracle Kernel handles best possible configuration to perform optimal I/O using available resources. This enables better configuration management.
- NFS storage is now available across different platforms like Windows.
- ODM NFS helps standardize all the tunable configuration parameters.
- ODM NFS has a stable NFS client that does not affect kernel performance. It optimizes the I/O path when making NFS operations.This ensures higher stability.
- Better diagnostics in case of errors.

Choosing Mount Points

This section provides guidelines on how to choose the mount points for the file systems that you want to use for the Oracle software and database files. The guidelines contained in the following sections comply with the Optimal Flexible Architecture recommendations:

- Choosing Mount Points for Oracle Software Files
- Choosing Mount Points for Oracle Database and Recovery Files

Choosing Mount Points for Oracle Software Files

Oracle software files are stored in three different directories:

- Oracle base directory
- Oracle Inventory directory
- Oracle home directory

For the first installation of Oracle software on a system, the Oracle base directory, identified by the `ORACLE_BASE` environment variable, is normally the parent directory for both the Oracle Inventory and Oracle home directories. For example, for a first installation, the Oracle base, Oracle Inventory, and Oracle home directories might have paths similar to the following:

Directory	Path
Oracle base (`$ORACLE_BASE`)	`/u01/app/oracle`
Oracle Inventory	`$ORACLE_BASE/oraInventory`
Oracle home	`$ORACLE_BASE/product/11.1.0/db_1`

For subsequent installations, you can choose to use either the same Oracle base directory or a different one, but every subsequent installation uses the original Oracle Inventory directory. For example, if you use the `/u02/app/oracle` directory as the Oracle base directory for a new installation, then the Oracle Inventory directory continues to be `/u01/app/oracle/oraInventory`.

To enable you to effectively maintain the Oracle software on a particular system, Oracle recommends that you locate the Oracle Inventory directory only on a local file system, if possible. If you must place the Oracle Inventory directory on a NAS device, create a specific directory for each system, then to prevent more than one system from writing to the same Inventory.

Directory-Specific Guidelines

You can use any of the following directories as mount points for NFS file systems used to store Oracle software:

> **Note:** In the following examples, the paths shown are the defaults if the ORACLE_BASE environment variable is set before you start Oracle Universal Installer.

- Oracle base directory or its parents (/u01/app/oracle for example)

  If you use the Oracle base directory of one of its parents as a mount point, then the default location for all Oracle software and database files will be on that file system. During the installation, you might consider changing the default location of the following directories:

  – The Oracle Inventory directory (*oracle_base*/oraInventory)

    Specify a local file system or a host-specific directory on the NFS file system, for example:

    oracle_base/*hostname*/oraInventory

  – The Oracle database file directory (*oracle_base*/oradata)

    You might want to use a different file system for database files, for example, to enable you to specify different mount options or to distribute I/O.

  – The Oracle database recovery file directory (*oracle_base*/flash_recovery_area)

    Oracle recommends that you use different file systems for database and recovery files.

  If you use this mount point, then all Oracle installations that use this Oracle base directory will use the NFS file system.

- The product directory (*oracle_base*/product)

  By default, only software files will be located on the NFS file system. You can also use this mount point to install software from different releases, for example:

  ```
  /u01/app/oracle/product/9.2.0
  /u01/app/oracle/product/10.2.0/db_1
  /u01/app/oracle/product/11.1.0/db_1
  ```

- The release directory (*oracle_base*/product/11.1.0)

  By default, only software files will be located on the NFS file system. You can also use this mount point to install different products from the same release, for example:

  ```
  /u01/app/oracle/product/11.1.0/db_1
  /u01/app/oracle/product/11.1.0/client_1
  ```

- The Oracle home directory (*oracle_base*/product/11.1.0/db_1)

  By default, only software files will be located on the NFS file system. This is the most restrictive mount point. You can use it only to install a single release of one product:

  ```
  /u01/app/oracle/product/11.1.0/db_1
  ```

Choosing Mount Points for Oracle Database and Recovery Files

To store Oracle database or recovery files on a NAS device, you can use different paths depending on whether you want to store files from only one database or from more than one database:

- Use the NFS file system for files from more than one database

  If you want to store the database files or recovery files from more than one database on the same NFS file systems, then use paths or mount points similar to the following:

File Type	Path or Mount Point
Database files	`/u02/oradata`
Recovery files	`/u03/flash_recovery_area`

  When Oracle Universal Installer prompts you for the data file and the recovery file directories, specify these paths. The Database Configuration Assistant and Enterprise Manager create subdirectories in these directories using the value you specify for the database name (DB_NAME) as the directory name, for example:

  ```
  /u02/oradata/db_name1
  /u03/flash_recovery_area/db_name1
  ```

- Use the NFS file system for files from only one database

  If you want to store the database files or recovery files for only one database in the NFS file system, then you can create mount points similar to the following, where `orcl` is the name that you want to use for the database:

  ```
  /u02/oradata/orcl
  /u03/flash_recovery_area/orcl
  ```

  Specify the directory `/u02/oradata` when Oracle Universal Installer prompts you for the data file directory and specify the directory `/u03/flash_recovery_area` when Oracle Universal Installer prompts you for the recovery file location. The `orcl` directory will be used automatically either by Database Configuration Assistant or by Enterprise Manager.

Creating Files on a NAS Device for Use with Automatic Storage Management

If you have a certified NAS storage device, then you can create zero-padded files in an NFS mounted directory and use those files as disk devices in an Automatic Storage Management disk group. To create these files, follow these steps:

> **Note:** To use files as disk devices in an Automatic Storage Management disk group, the files must be on an NFS mounted file system. You cannot use files on local file systems.

1. If necessary, create an exported directory for the disk group files on the NAS device.

   Refer to the NAS device documentation for more information about completing this step.

2. Switch user to `root`:

```
$ sudo sh
password:
```

3. Create a mount point directory on the local system:

```
# mkdir -p /mnt/oracleasm
```

4. To ensure that the NFS file system is mounted when the system reboots, add an entry for the file system in the `/etc/fstab` mount file.

   For more information about editing the mount file for the operating system, refer to the man pages. For more information about recommended mount options, refer to the "NFS Mount Options" section on page C-5.

5. Enter a command similar to the following to mount the NFS file system on the local system:

```
# mount /mnt/oracleasm
```

6. Choose a name for the disk group that you want to create, for example, `nfsdg`.

7. Create a directory for the files on the NFS file system, using the disk group name as the directory name:

```
# mkdir /mnt/oracleasm/nfsdg
```

8. Use commands similar to the following to create the required number of zero-padded files in this directory:

```
# dd if=/dev/zero of=/mnt/oracleasm/nfsdg/disk1 bs=1024k count=1000
```

   This example creates 1 GB files on the NFS file system. You must create one, two, or three files respectively to create an external, normal, or high redundancy disk group.

   Note: Creating multiple zero-padded files on the same NAS box does not guard against NAS box failure. Instead, create one file for each NAS box and mirror across them using the Automatic Storage Management technology.

9. Enter the following commands to change the owner, group, and permissions on the directory and files that you created:

```
# chown -R oracle:dba /mnt/oracleasm
# chmod -R 660 /mnt/oracleasm
```

10. When you are creating the database, edit the Automatic Storage Management disk discovery string to specify a regular expression that matches the file names you created. For example, you might specify a disk discovery string similar to the following:

```
/mnt/oracleasm/nfsdg/*
```

NFS Mount Options

You must mount NFS volumes used for storing database files with special mount options on the host where the database server is running. When mounting an NFS file system, Oracle recommends that you use the same mount point options that the NAS

vendor used when certifying the device. Refer to the device documentation or contact the vendor for information about recommended mount-point options.

Option	Requirement	Description
`hard`	Mandatory	Generate a hard mount of the NFS file system. If the connection to the server fails or is temporarily lost, then connection attempts are made until the NAS device responds.
`bg`	Optional	Try to connect in the background if connection fails.
`tcp`	Optional	Use the TCP protocol rather than UDP. TCP is more reliable than UDP.
`nfsvers=3`	Optional	Use NFS version 3. Oracle recommends that you use NFS version 3 where available, unless the performance of version 2 is later.
`suid`	Optional	Allow clients to run executables with SUID enabled. This option is required for Oracle software mount points.
`rsize`	Mandatory	The number of bytes used when reading from the NAS device. This value should be set to the maximum database block size supported by this platform. A value of 8192 is often recommended for NFS version 2 and 32768 is often recommended for NFS version 3.
`wsize`	Mandatory	The number of bytes used when writing to the NAS device. This value should be set to the maximum database block size supported by this platform. A value of 8192 is often recommended for NFS version 2 and 32768 is often recommended for NFS version 3.
`nointr` (or `intr`)	Optional	Do not allow (or allow) keyboard interrupts to stop a process that is hung while waiting for a response on a hard-mounted file system. **Note:** Different vendors have different recommendations about this option. Contact the vendor for advice.
`actimeo=0` or `noac`	Mandatory	Disable attribute caching. **Note:** You must specify this option for NFS file systems where you want to install the software. If you do not use this option, then Oracle Universal Installer will not install the software in the directory that you specify.
`directio`	Optional	Disable attribute caching. **Note**: If the systems supports directio, use this option instead of `noac` to reliably disable caching.

D

Optimal Flexible Architecture

This appendix describes the Optimal Flexible Architecture standard. The standard is a set of configuration guidelines created to ensure well organized Oracle installations that are easier to maintain. It includes information about the following topics:

- Overview of the Optimal Flexible Architecture Standard
- Implementing Optimal Flexible Architecture

Overview of the Optimal Flexible Architecture Standard

All Oracle components on the installation media are compliant with Optimal Flexible Architecture, which means that 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 the database will grow in size, or if you plan to have multiple databases.

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
- Exploiting the Optimal Flexible Architecture Structure for Oracle Files
- Optimal Flexible Architecture File Mapping

File Systems

The following sections describe the conventions for mount points:

- Number of File Systems
- Naming Conventions

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 Directories

The following sections describe the naming conventions for directories that are compliant with the Optimal Flexible Architecture standard:

- Oracle Base Directory Naming Convention
- Naming Mount Points for Very Large Databases (VLDBs)
- Referring to Path Names
- Oracle Home Directory Naming Convention
- Naming Subdirectories

Oracle Base Directory Naming Convention

The Oracle Base directory is the top level directory that you can use to install the various oracle software products. You can use the same Oracle base directory for more than one installation. If different operating system users install Oracle software on the same system, then each user must create a separate Oracle base directory.

Name Oracle base directories using the syntax */pm/s/u*. Table D–1 describes the variables used in this syntax.

Table D–1 Syntax for Naming Oracle Base Directories

Variable	Description
pm	A mount point name
s	A standard directory name
u	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/*`.

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, use the syntax */h/q/d* for naming mount points. Table D–2 describes the variables used in this syntax.

Table D–2 Syntax for Naming Mount Points for Very Large Databases

Variable	Description
h	Oracle base directory
q	A string denoting that Oracle data is stored in this directory, for example, `oradata`
d	The value of the initialization parameter `DB_NAME` (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`.

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 D–3 describes the variables used in this syntax.

Table D–3 Syntax for Naming Oracle Home Directories

Variable	Description
pm	A mount point name
s	A standard directory name
u	The name of the owner of the directory
v	The version of the software
type	The type of installation, for example Database (`db`), Client (`client`), or Oracle Clusterware (`crs`)
n	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/11.1.0/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 D–4 describes the database administration file subdirectories.

Table D–4 Subdirectories for Database Administration Files

Subdirectory	Description
adhoc	Ad hoc SQL scripts
arch	Archived redo log files
adump	Audit files (Set the AUDIT_FILE_DEST initialization parameter to specify the adump directory. Clean out this subdirectory periodically.)
create	Scripts used to create the database
exp	Database export files
logbook	Files recording the status and history of the database
pfile	Instance parameter files

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

In Oracle Database 11*g*, Automatic Diagnostic Repository (ADR) directories replace the bdump, cdump, and udump directories. The ADR diagnostic data will go into the /h/diag/rdbms/d/i/ directory.

where

h is Oracle Base

d is the database name

i is the instance name.

From there we have the trace, alert, and incident sub-directories.

Table D–5 Locations for Diagnostic Traces

Diagnostic Data	10*g* Location	11*g* Location
Foreground Process traces	user_dump_dest	{ADR_HOME}/trace/
Background Process traces	background_dump_dest	{ADR_HOME}/trace/
Alert Log Data	background_dump_dest	{ADR_HOME}/alert/
Core Dump	core_dump_dest	{ADR_ HOME}/incident/In/
Incident Dumps	user_dump_dest or background_dump_dest depending on the process.	{ADR_ HOME}/incident/In/

Naming Database Files

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

Note: Oracle Managed Files (OMF) and files stored in Automatic Storage Management disk groups 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	/*h*/*q*/*d*/control.ctl

File Type	File Naming Convention
Redo log files	/*h*/*q*/*d*/redo*n*.log
Data files	/*h*/*q*/*d*/*tn*.dbf

The following table describes this syntax:

Variable	Description
h	Oracle base directory
q	A string (typically oradata) distinguishing Oracle data from all other files
d	The value of the DB_NAME initialization parameter (typically, the same as the instance SID for single-instance databases)
t	An Oracle tablespace name
n	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 /*h*/*q*/*d*.

Using this convention, it is easy to determine the database to which the /u01/app/oracle/oradata/sab/system01.dbf file belongs.

Separating Segments with Different Requirements

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

Table D–6 describes the special tablespaces that the Database Configuration Assistant creates for each Oracle database. If you manually create a database, 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 D–6 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.

Exploiting the Optimal Flexible Architecture Structure for Oracle Files

Table D–7 describes the syntax used for identifying classes of files.

Table D–7 Directory Structure Syntax for Identifying Classes of Files

Directory Structure Syntax	Description
`/u[0-9][0-9]`	User data directories
`/*/home/*`	User home directories
`/*/app/*`	User application software directories
`/*/app/applmgr`	Oracle applications software subtrees
`/*/app/oracle/product`	Oracle software subtrees
`/*/app/oracle/product/11.1.0`	Oracle software subtree for release 11*g* products
`/*/app/oracle/product/11.1.0/db*`	Oracle home directories for Oracle Database 11*g*
`/*/app/11.1.0/crs`	Oracle home directory for Oracle Clusterware 11*g*
`/*/app/oracle/admin/orcl`	`orcl` database administrative subtrees
`/*/app/oracle/admin/orcl/arch/*`	`orcl` database archived log files
`/*/app/oracle/oradata`	Oracle data directories
`/*/app/oracle/oradata/orcl/*`	`orcl` database files
`/*/app/oracle/oradata/orcl/*.log`	`orcl` database redo log files

Optimal Flexible Architecture File Mapping

Table D–8 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`.

Note: Oracle recommends that you use ASM to provide greater redundancy and throughput.

Table D–8 Hierarchical File Mapping for an Optimal Flexible Architecture Installation

Directory	Description
`/`	Root directory
`/u01/`	User data mount point 1
`/u01/app/`	Subtree for application software
`/u01/app/oracle/`	Oracle Base directory
`/u01/app/oracle/admin/`	Subtree for database administration files
`/u01/app/oracle/admin/TAR`	Subtree for support log files
`/u01/app/oracle/admin/`*`db_name1`*`/`	`admin` subtree for *db_name1* database
`/u01/app/oracle/admin/`*`db_name2`*`/`	`admin` subtree for *db_name2* database
`/u01/app/oracle/doc/`	Online documentation
`/u01/app/oracle/flash_recovery_area/`	Subtree for recovery files
`/u01/app/oracle/flash_recovery_area/`*`db_name1`*	Recovery files for *db_name1* database

Table D–8 (Cont.) Hierarchical File Mapping for an Optimal Flexible Architecture Installation

Directory	Description
`/u01/app/oracle/flash_recovery_area/`*`db_name2`*	Recovery files for *db_name2* database
`/u02/app/oracle/oradata`	Oracle data directory
`/u03/app/oracle/oradata`	
`/u04/app/oracle/oradata`	
`/u01/app/oracle/product/`	Distribution files
`/u01/app/oracle/product/11.1.0/db_1`	Oracle home directory for Oracle Database 11*g*
`/u01/app/11.1.0/crs`	Oracle home directory for Oracle Clusterware11*g*
`/u01/app/kjf/`	Oracle base directory for user kjf
`/u01/app/edm/`	Oracle base directory for user edm

E

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

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 make sure 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`, 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 E–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
Oracle Management Agent HTTP port for Enterprise Management Agent. It is configured during installation. "Changing the Oracle Enterprise Management Agent Port" on page E-4 explains how to modify its port number.	3938	1830–1849	HTTP
Oracle Enterprise Manager Database Console HTTP port for Enterprise Manager Database Control. It is configured during installation. "Changing the Oracle Enterprise Manager Database Console Ports" on page E-4 explains how to modify its port number.	1158	5500–5519	TCP/HTTP
Oracle Enterprise Manager Database Console RMI port for Enterprise Manager Database Control. It is configured during installation."Changing the Oracle Enterprise Manager Database Console Ports" on page E-4 explains how to modify its port number.	5520	5520–5539	TCP

Table E–1 (Cont.) Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol
Oracle Enterprise Manager Database Console JMS port for Enterprise Manager Database Control. It is configured during installation. "Changing the Oracle Enterprise Manager Database Console Ports" on page E-4 explains how to modify its port number.	5540	5540–5559	TCP
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 E-5 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 E-5 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 E-5 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 need to 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 E-5 explains how to change its port number.	Dynamic	Dynamic	HTTP
Oracle XML DB The Oracle XML DB FTP is used when applications need to 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 E-5 explains how to change its port number.	Dynamic	Dynamic	FTP
Oracle RAC (UNIX) The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	UDP
Oracle Clusterware Oracle Clusterware Daemon internode connection. The port number is assigned automatically during installation. You cannot view or modify it afterward.	49896	49896	TCP
Cluster Synchronization Service (CSS) CSS daemon internode connection for the GM layer. The port number is assigned automatically during installation. You cannot view or modify it afterward.	49895	49895	TCP

Table E–1 (Cont.) Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol
Oracle Cluster Registry The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	TCP
Oracle Event Manager The port number is assigned automatically during installation. You cannot view or modify it afterward.	49897	49897–49898	TCP
Cluster Manager The port number is assigned automatically during installation. You cannot view or modify it afterward.	Dynamic	Dynamic	TCP

Changing the Oracle Enterprise Management Agent Port

To find the current setting for the Oracle Management agent port, search for `EMD_URL` in the `$ORACLE_HOME/`*`host_sid`*`/sysman/config/emd.properties` file.

To change the Oracle Management Agent HTTP port, use the `emca -reconfig` ports command:

```
emca -reconfig ports -AGENT_PORT 1831
```

Changing the Oracle Enterprise Manager Database Console Ports

To find the current HTTP, RMI, and JMS port settings, search in the following files:

- HTTP port: Search for `REPOSITORY_URL` in the `$ORACLE_HOME/`*`host_sid`*`/sysman/config/emd.properties` file.
- RMI port: Search for the `port` attribute in the `rmi-server` tag in the `$ORACLE_HOME/oc4j/j2ee/OC4J_DBConsole_`*`host_sid`*`/config/rmi.xml` file.
- JMS port: Search for the `port` attribute in the `jms-server` tag in the `$ORACLE_HOME/oc4j/j2ee/OC4J_DBConsole_`*`host_sid`*`/config/jms.xml` file.

To change the Oracle Enterprise Manager Database Control ports, use the `emca -reconfig ports` command:

```
$ORACLE_HOME/bin> emca -reconfig ports option setting
```

where *`option`* can be:

- `DBCONTROL_HTTP_PORT`: Sets the HTTP port, for example:

  ```
  emca -reconfig ports -DBCONTROL_HTTP_PORT 1820
  ```

- `RMI_PORT`: Sets the RMI port, for example:

  ```
  emca -reconfig ports -RMI_PORT 5520
  ```

- `JMS_PORT`: Sets the JMS port, for example:

  ```
  emca -reconfig ports -JMS_PORT 5521
  ```

You can enter multiple `-reconfig port` settings in one line, for example:

```
emca -reconfig ports -DBCONTROL_HTTP_PORT 1820 -AGENT_PORT 1821 -RMI_PORT 5520
```

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 need to 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, make sure that the Oracle TNS Listener service (for example, `OracleOraDb11g_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 as `SYS` or `XDB` using the `SYSDBA` privilege.

   For example, to log in to SQL*Plus as `SYS` using the password `welcome`:

   ```
   $ $ORACLE_HOME/bin/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 the Oracle home is in the following location, enter the following command:

   ```
   SQL> $ORACLE_HOME/rdbms/admin/catxdbdbca.sql 2200 8200
   ```

4. Exit SQL*Plus.

F

Configuring Oracle Database Globalization Support

This appendix describes the following Globalization Support topics:

- Installing and Using Oracle Components in Different Languages
- Running Oracle Universal Installer in Different Languages

Installing and Using Oracle Components in Different Languages

This section describes the following procedures:

- Configuring Oracle Components to Run in Different Languages
- Installing Translation Resources

Configuring Oracle Components to Run in Different Languages

You can specify the language and the territory, or locale, in which you want to use Oracle components. The locale setting of a component determines the language of the user interface of the component and the globalization behavior, such as date and number formatting. Depending on the Oracle component, the locale of the component is either inherited from the operating system session that started the component, or is defined by the `NLS_LANG` environment variable.

The operating system locale usually influences Oracle components that are based on Java technology. The `NLS_LANG` environment variable usually influences Oracle components that use Oracle Client libraries such as OCI.

> **Note:** The user interface of an Oracle component will be displayed in a selected language only if the appropriate translation is available and has been installed. Otherwise, the user interface will be displayed in English.

This section describes the following procedures:

- Determining the Operating System Locale by Using the LANG Environment Variable
- Configuring Locale and Character Sets by Using the NLS_LANG Environment Variable

Determining the Operating System Locale by Using the LANG Environment Variable

The locale setting of your operating system session determines the language of the user interface and the globalization behavior for components such as Oracle Universal Installer, Oracle Net Configuration Assistant, and Oracle Database Configuration Assistant. It also determines the globalization behavior of Oracle Database sessions created by a user application through Oracle JDBC driver, unless overridden by the application.

The operating system locale on Linux is determined by the value of the `LANG` environment variable. Depending on your desktop environment, such as KDE, GNOME, or telnet, you can select a default session locale on a login screen, in a configuration panel, or in a configuration file.

> **Note:** Refer to the operating system documentation on how to select a locale for the operating system session in your desktop environment.

You can modify the `LANG` variable in the environment of your shell to start an Oracle component in a selected language. For example, to start Oracle Database Configuration Assistant in German, enter one of the following commands:

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

```
$ LANG=de_DE.iso88591 dbca
```

- C shell (csh):

```
% (setenv LANG de_DE.iso88591; dbca)
```

> **Note:** The `LC_ALL` environment variable overrides the value of the LANG environment variable. For the commands listed in the following section to work, either ensure that the `LC_ALL` environment variable is not set in the environment, or substitute `LC_ALL` for `LANG`.

To modify the operating system locale for all Oracle components started from now on by the given shell, modify the `LANG` variable using one of the following commands:

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

```
$ LANG=de_DE.iso88591; export LANG
$ ...
```

- C shell (csh):

```
% setenv LANG de_DE.iso88591
$ ...
```

The value of the `LANG` environment variable must be a valid operating system locale. To see the list of valid locales, enter the following command:

```
$ locale -a
```

> **Note:** Refer to the operating system documentation for a mapping between values of the `LANG` environment variable and the languages and territories that they represent.

Configuring Locale and Character Sets by Using the NLS_LANG Environment Variable

The NLS_LANG environment variable determines the language of the user interface and the globalization behavior for components such as SQL*Plus, exp, and imp. It sets the language and territory used by the client application and the database. It also declares the character set for entering and displaying data by the client application.

The NLS_LANG environment variable uses the following format:

```
NLS_LANG=language_territory.characterset
```

In this format:

- *language* specifies the language used for displaying Oracle messages, sorting, day names, and month names
- *territory* specifies the conventions for default date, monetary and numeric formats
- *characterset* specifies the encoding used by the client application

  In most cases, this is 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 value of NLS_LANG environment variable is AMERICAN_AMERICA.AL32UTF8.

> **See Also:** *Oracle Database Globalization Support Guide* for information about the NLS_LANG parameter and Globalization Support initialization parameters

The following examples illustrate some of the valid values for the NLS_LANG environment variable.

> **Note:** Refer to the operating system 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

Installing Translation Resources

To view the user interface of Oracle components in different languages, you must install the appropriate language translations along with the component.

> **Note:** Part of Oracle Database Vault user interface text is stored in database tables in the DVSYS schema. By default, only the English language is loaded into these tables. You can use Oracle Database Vault Configuration Assistant to add more languages to Oracle Database Vault. For the necessary steps, refer to Appendix C in *Oracle Database Vault Administrator's Guide*.

To select the translation resources that you want to install:

1. Start Oracle Universal Installer.
2. On the Select Installation Method screen, select Advanced Installation and click **Next**.
3. On the Select Installation Type screen, click **Product Languages**.
4. On the Language Selection screen, select the language in which you want to use Oracle components from the Available Languages field.

> **Note:** The Available Languages field lists all languages supported by Oracle globalization libraries. The set of languages for which a translation is actually available is usually smaller and depends on a particular component. The scope of translation for a given component may differ between languages. For example, some translations may include all user interface text, while others may include only error messages and no help files.

5. Use the **>** arrow to move the selected language to the Selected Languages field, and then click **OK**.

> **Note:** Oracle Universal Installer will ignore languages in the Selected Languages field for which no translation is available.

6. Select the installation type you want, and then click **Next**.

> **Note:** To install additional languages for a component, you will have to reinstall this component.

Running Oracle Universal Installer in Different Languages

Your operating system locale determines the language in which Oracle Universal Installer runs. Oracle Universal Installer may run in one of the following languages:

- Brazilian Portuguese (pt_BR)
- French (fr)
- German (de)
- Italian (it)
- Japanese (ja)
- Korean (ko)
- Simplified Chinese (zh_CN)

- Spanish (es)
- Traditional Chinese (zh_TW)

To run Oracle Universal Installer in one of the available languages, change the locale in which your operating system session is running before you start Oracle Universal Installer with the `./runInstaller` command. If the selected language is not one of them listed earlier, Oracle Universal Installer runs in English.

You need to ensure that the selected value for the `LANG` environment variable starts with the appropriate language abbreviation. In the aforementioned list of languages, in which Oracle Universal Installer can run, the required abbreviation appears in parentheses beside the language name. For example, fr_FR and fr_CA are valid values to run the Oracle Universal Installer in French.

G

Troubleshooting

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

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 11*g* DVD. The latest version of the release notes is also available on the Oracle Technology Network Web site:

`http://www.oracle.com/technology/documentation/`

X Window Display Errors

If you run Oracle Universal Installer on a remote system and you want to display Oracle Universal Installer's user interface on your local system, you might see error messages similar to the following:

```
"Failed to connect to server"
"Connection refused by server"
```

```
"Can't open display"
```

If you see one of these error messages, follow these steps:

> **Note:** This procedure applies only to users of UNIX workstations. If you are using a PC or other system with X server software installed, refer to the X server documentation for information about how to permit remote systems to display X applications on the local system.

1. In a local terminal window, log in as the user that started the X Window session.
2. Enter the following command:

   ```
   $ xhost fully_qualified_remote_host_name
   ```

   For example:

   ```
   $ xhost somehost.us.example.com
   ```

3. Enter the following commands, where *workstation_name* is the host name or IP address of your workstation:

   - Bourne, Bash, or Korn shell:

     ```
     $ DISPLAY=workstation_name:0.0
     $ export DISPLAY
     ```

   - C shell:

     ```
     % setenv DISPLAY workstation_name:0.0
     ```

4. To determine whether X Window applications display correctly on the local system, enter the following command:

   ```
   $ xclock
   ```

   The X clock should appear on your monitor.

5. If the X clock appears, close the X clock and start Oracle Universal Installer again.

What to Do If an Installation Error Occurs?

If you encounter an error during installation:

- Do not exit Oracle Universal Installer.
- If you clicked **Next** after you entered incorrect information on one of the installation screens, click **Back** to return to the screen and correct the information.
- If you encounter an error while Oracle Universal Installer is copying or linking files, refer to "Reviewing the Log of an Installation Session" section on page G-3.
- If you encounter an error while a configuration assistant is running, refer to "Troubleshooting Configuration Assistants" section on page G-6.
- If you cannot resolve the problem, remove the failed installation by following the steps listed in the "Cleaning Up After a Failed Installation" section on page G-7.

Reviewing the Log of an Installation Session

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

> **Note:** If you run Oracle Universal Installer during the time that daily cron jobs run, then you may encounter unexplained installation problems if your cron job is performing cleanup, and temporary files are deleted before the installation is finished. Oracle recommends that you complete installation before daily cron jobs are run, or disable daily cron jobs that perform cleanup until after the installation is completed.

To view the log file, follow these steps:

1. If necessary, enter the following command to determine the location of the `oraInventory` directory:

   ```
   $ cat /etc/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 Hostname Changes and CSS

If you change the host name for ASM, then the Oracle CSS daemon will not start. In order to counter this problem, please use the following steps:

- Login as the `root` user

- Run `localconfig delete` to deconfigure CSS. This will remove any configuration related files on the system that referenced the old host name.
- Run `localconfig add` to reconfigure CSS using the new host name.

For Example:

```
# $ORACLE_HOME/bin/localconfig [add] [delete] [ reset destination_Oracle_home ]
[-silent] [-paramfile Complete_path_of_file_specifying_parameter_values]
```

Troubleshooting Oracle Configuration Manager

This section lists some of the errors that may occur while using Oracle Configuration Manager and provides tips to troubleshoot these errors.

- Insufficient Privileges While Running installCCRSQL collectconfig

  When you run the `installCCRSQL.sh` script, it creates the `ORACLE_OCM` user and sets up a job to collect database configuration information. The `ORACLE_OCM` user requires `EXECUTE` privileges on `UTL_FILE` and `DBMS_SCHEDULER` for database versions 10g or later, and on the `DBMS_JOB` for pre-10*g* databases. If these privileges are granted to `PUBLIC`, the `ORACLE_OCM` user inherits these privileges, otherwise these privileges are explicitly granted when the `installCCRSQL.sh` script is executed. If the inherited privileges are revoked, the following errors indicating the lack of privileges will be logged in the `alert_log`:

  ```
  ORA-12012: error on auto execute of job 52
  ORA-04068: existing state of packages has been discarded
  ORA-04063: package body "ORACLE_OCM.package_name" has errors
  ORA-06508: PL/SQL: could not find program unit being called
  ```

  To resolve these errors, you must grant the missing `EXECUTE` privilege to the `ORACLE_OCM` user.

  – For database versions 10g and later, grant `EXECUTE` privileges on the `UTL_FILE` and `DBMS_SCHEDULER` packages to the `ORACLE_OCM` user by entering the following `SQL*PLUS` commands:

    ```
    SQL> grant execute on UTL_FILE to oracle_ocm;
    SQL> grant execute on DBMS_SCHEDULER to oracle_ocm;
    SQL> ALTER PACKAGE oracle_ocm.MGMT_DB_LL_METRICS compile;
    SQL> ALTER PACKAGE oracle_ocm.mgmt_config compile;
    ```

  – For pre-10g databases, grant `EXECUTE` privileges on the `DBMS_JOB` package to the `ORACLE_OCM` user by entering the following `SQL*PLUS` commands:

    ```
    SQL> grant execute on UTL_FILE to oracle_ocm;
    SQL> grant execute on DBMS_JOB to oracle_ocm;
    SQL> ALTER PACKAGE oracle_ocm.MGMT_DB_LL_METRICS compile;
    SQL> ALTER PACKAGE oracle_ocm.mgmt_config compile;
    ```

- ORA-04021 Error

  There may be cases when the `ORACLE_OCM` user must be granted the required privileges during installation. While granting the privileges, the following error may occur in the `$ORACLE_HOME/ccr/log/collectconfig`*`SID`*`.log`:

  ```
  ORA-04021: timeout occurred while waiting to lock object SYS.<package like UTL_
  FILE>
  ```

  This error may occur if another procedure is using the package for which the privileges are being granted. To resolve this error, retry the install when the

package is not being used. This error may occur while granting privileges on UTL_FILE, DBMS_SCHEDULER, or DBMS_JOB.

- ORA-01925 Error While Running installCCRSQL

  This error may occur if the value of the MAX_ENABLED_ROLES initialization parameter has been exceeded. To resolve this error, you must increase the value of the MAX_ENABLED_ROLES parameter and restart the database as follows:

  1. Edit the init*sid*.ora file where *sid* is the database system identifier and increase the value of MAX_ENABLED_ROLES . If a server parameter (spfile) has been used, alter the MAX_ENABLED_ROLES parameter by using the following SQL*PLus command:

     SQL>alter system set MAX_ENABLED_ROLES=*value* scope=spfile

  2. Restart the database.

  Once the database has been restarted, re-run the installCCRSQL.sh script.

- Incorrectly configured host names are displayed on Oracle*MetaLink* with only the short names

  To ensure that host names are displayed with their fully qualified names on Oracle*MetaLink*, the /etc/hosts file must contain an entry that includes both the host name and the domain in the following format:

  ```
  IP-Address  Full-HostName  Short-HostName
  ```

  For example:

  ```
  10.10.10.10  myhost.mydomain  myhost
  ```

  If the /etc/hosts file has not been correctly configured, only the short name is displayed on Oracle*MetaLink*.

- Oracle Configuration Manager Synchronization Messages: Oracle Configuration Manager does not allow you to run multiple commands simultaneously. If you attempt to do so, the following messages may be displayed:

  – Message: Another operation is in progress. Please wait...

    Description: There are several Oracle Configuration Manager commands that cannot run concurrently. If you try to run one of these commands while another command is in progress, the second command will not be executed until the first command is completed. A message indicating that another command is in progress is displayed. The second command will automatically be run when the first command is completed.

    Commands: emCCR collect, emCCR getupdates, emCCR update_components, and emCCR upload

    Action: Initially, take no action, the second command will be executed when the first command is completed. But if the command execution takes too long, a timeout will occur. If a timeout occurs, ensure there is no Oracle Configuration Manager activity by executing emCCR stop command. Delete the ccr/state/collector.lock file and restart the **Scheduler** by running the emCCR start command. If you run the command in **Disconnected** mode, ensure that no collection or update is taking place and then delete the ccr/state/collector.lock file.

  – Message: Operation blocked, waiting..

Description: You cannot run the `emCCR update_components` command if any other `emCCR` command is running. If you try to run the command, it will be blocked. You also cannot run any `emCCR` command while `emCCR update_components` is running as all other commands will be blocked.

Commands: `configCCR` and most of the `emCCR` commands

Action: Initially, take no action, the command will get executed when the current command is completed. If a timeout occurs, ensure that there is no Oracle Configuration Manager activity by executing `emCCR stop`. Delete the `ccr/state/semaphore.op*` and `ccr/state/semaphore.update*` files, and restart Oracle Configuration Manager by running `emCCR start`. If running the command in **Disconnected** mode, ensure no collection or update is taking place and delete the `ccr/state/semaphore.op*` and the `ccr/state/semaphore.update*` files.

– Message: The Scheduler is down for upgrade.

Description: While upgrading Oracle Configuration Manager, you cannot run any of the `emCCR` commands.

Commands: All `emCCR` commands

Action: Retry the commands later.

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 G-3.
- Review the specific configuration assistant log file located in the `$ORACLE_HOME/cfgtoollogs` directory. Try to fix the issue that caused the error.
- If you see the "Fatal Error. Reinstall" message, look for the cause of the problem by reviewing the log files. Refer to "Fatal Errors" on page G-6 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/installActions`*date_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 canceled	-1

Fatal Errors

If you receive a irrecoverable error while a configuration assistant is running, you must remove the current installation and reinstall the Oracle software, as follows:

1. Remove the failed installation as described in the "Cleaning Up After a Failed Installation" section on page G-7.
2. Correct the cause of the irrecoverable 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/silentInstall`*`date_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, check the `FROM_LOCATION` variable and make sure that it points to the `products.xml` file in the installation media. In the installation media, this `products.xml` is in `response/stage`.
- Oracle Universal Installer encounters an error, such as insufficient disk space

Oracle Universal Installer or configuration assistant validates the response file at run time. If the validation fails, the silent-mode installation or configuration process ends. Oracle Universal Installer treats values for parameters that are of the wrong context, format, or type as if no value was specified in the file.

Cleaning Up After a Failed Installation

If an installation fails, you must remove files that Oracle Universal Installer created during the attempted installation and remove the Oracle home directory. Perform the following steps to remove the files:

1. Start Oracle Universal Installer as described in the "Installing the Oracle Database Software" on page 3-7.
2. Click **Deinstall Products** on the Welcome window or click **Installed Products** on any Installer window.

   The Inventory window appears, listing installed products.
3. Select the Oracle home that contains the products that you want to remove, then click **Remove**.
4. Manually remove the Oracle home directory created during the failed installation.
5. Reinstall the Oracle software.

To reinstall, you need to drop either one or two database schemas, depending upon the installation type.

After Failed Upgrade Installation

In the case of a failed upgrade installation, you need to revert Oracle Application Express to a earlier release and then remove the schemas associated with release 2.2. This section contains the following topics:

- Reverting to Earlier Release
- After a Failed New Installation

Reverting to Earlier Release

To revert to a previous Oracle Application Express release:

1. If you altered the images directory, you need to point the text alias `/i/` back to images directory for release 1.5.
2. Run the following command in SQL*Plus:
   a. Start SQL*Plus and connect the database where Oracle Application Express is installed as `SYS`. For example:

```
$ $ORACLE_HOME/bin/sqlplus
SQL> CONNECT SYS as SYSDBA
Enter password: SYS_password
```

   b. To revert to Oracle Application Express release 1.5, execute the following:

```
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_010500;
exec flows_010500.wwv_flow_upgrade.switch_schemas
('FLOWS_030000','FLOWS_010500');
```

   c. To revert to Oracle Application Express release 1.6, execute the following:

```
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_010600;
exec flows_010600.wwv_flow_upgrade.switch_schemas
('FLOWS_030000','FLOWS_010600');
```

   d. To revert to Oracle Application Express release 2.0, execute the following:

```
ALTER SESSION SET CURRENT_SCHEMA = FLOWS_020000;
exec flows_020000.wwv_flow_upgrade.switch_schemas
('FLOWS_030000','FLOWS_020000');
```

To remove the release 2.2 schema:

1. Start SQL*Plus and connect the database where Oracle Application Express is installed as `SYS`.
2. Execute the following commands:

```
DROP user FLOWS_030000 CASCADE;
```

After a Failed New Installation

To remove schemas after a failed new installation:

1. Start SQL*Plus and connect the database where Oracle Application Express is installed as `SYS`.
2. Run the following commands:

```
drop user FLOWS_030000 cascade;
drop user FLOWS_FILES cascade;
```

Images Displaying Incorrectly in Oracle Application Express

In Configuring Oracle HTTP Server in a New Installation on page 4-18, you added an alias entry that points to the file system path where you copied the images directory. If images in Oracle Application Express do not display correctly, you may have more than one definition of the `/i/` alias. To address this issue:

- If possible, rename the first instance of `/i/` to a different alias name.
- Alternatively, copy the images from the `$ORACLE_HOME/marvel/images` directory to the directory defined by the first `/i/` alias.

Online Help Not Working

If users are accessing Oracle Application Express through a Virtual Host, online Help will not work. Consider the following example:

- The hostname of the Oracle HTTP Server where the Oracle Application Express DAD resides is `internal.server.com` and the port is `7777`.
- Users access Oracle Application Express through a Virtual Host. In their Web browsers, users see `external.server.com` and port `80`.

In this example, Oracle Application Express online Help will not work if the users cannot access `internal.server.com`. To resolve this issue, add the following lines to the Oracle Application Express Database Access Descriptor (DAD) to override the CGI environment variables `SERVER_NAME` and `SERVER_PORT`:

```
PlsqlCGIEnvironmentList SERVER_NAME=external.server.com
PlsqlCGIEnvironmentList SERVER_PORT=80
```

> **See Also:** *Oracle Application Server mod_plsql User's Guide* for information on overriding the CGI environment variables.

H

Frequently Asked Questions About Installation

Use the following guidelines to decide how to install Oracle Database components:

- Installing Oracle Database or Oracle Database
- Installing Oracle Database Tools
- Installing Oracle Database with Oracle Applications
- Installing Oracle Database Heterogeneous Connectivity Tools (Gateways)

> **Note:** Some Oracle Database components may not be available on all platforms. Consult your platform-specific installation guide or release notes.

Installing Oracle Database or Oracle Database

The following are frequently asked questions with respect to installing 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?
- 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 Client if my client nodes have limited disk space?
- How do I upgrade Oracle Database?
- The computers at my site have been configured to run as a cluster. How should I install 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?

- If you want a quick installation using the default installation settings, then refer to the platform-specific *Oracle Database Quick Installation Guide*.
- If your site has special requirements, then refer to this guide for more information.

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 this guide for more details. 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 User's Guide*
- *Oracle OLAP DML Reference*
- *Oracle OLAP Java API Reference*

What's the best way to install multiple Oracle databases?

Use this guide 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, which is described in *Oracle Database Administrator's Guide*.

How do I configure client connections to an Oracle database?

1. Install Oracle Database on a server by using this guide for more information.
2. Use platform-specific *Oracle Database Client Installation Guide* to install Oracle Client on each client node, and select the Instant Client installation type.

   If you have many client nodes, consider staging the software centrally, mapping the drive, and running Oracle Universal Installer in the noninteractive mode.

   If the client nodes only require a default installation into a new Oracle home directory, consider using this guide for more information.

What is the best way to install Oracle Client if my client nodes have limited disk space?

1. Install Oracle Database onto a server by using this guide for more details.

2. Use platform-specific *Oracle Database Client Installation Guide* to install Oracle 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

The computers at my site have been configured to run as a cluster. How should I install Oracle Database?

Use any of the following installation scenarios:

- If you want to run a single-instance Oracle Database in a clustered environment, then install Oracle Clusterware either before or after you install Oracle Database.
- If you want a consolidated pool of storage for all databases in a cluster, then install Oracle Clusterware first and use Automatic Storage Management to manage this storage. Afterwards, install Oracle Database (which can be either single instance or Real Application Clusters).
- If you plan to use Oracle Real Application Clusters, first install Oracle Clusterware, and then install Oracle Real Application Clusters.

Refer to platform-specific *Oracle Clusterware Installation Guide* and *Oracle Real Application Clusters Installation Guide for Linux and UNIX* for the platform to install Oracle Clusterware or Oracle Real Application Clusters. Oracle Clusterware is available on the Oracle Clusterware installation media. Refer to this guide which explains how to install Automatic Storage Management and Oracle Database.

Oracle Clusterware is a key component required by Oracle Real Application Clusters installations. Oracle Clusterware is an integrated cluster management solution that can bind multiple servers together to act as a single system. This is referred to as a cluster. It performs workload management and component restart. For example, when an instance supporting a particular service fails, Oracle Clusterware restarts the service on the next available instance that you have configured for that service. Oracle Clusterware can monitor non-Oracle programs, as long as they are defined within the Oracle Clusterware environment using the High Availability API.

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

The following are frequently asked questions with respect to 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?

- How do I use Oracle Database to manage my XML data?
- 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?
- Is Oracle Workflow included with Oracle Database 11g?
- Is there a migration plan for customers that have built solutions using Oracle Workflow?

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 a separate Oracle Application Server instance. This database is populated with the metadata that Oracle Application Server must run.
- 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 this guide to install Oracle Database.
2. From Oracle Database, use Database Configuration Assistant to manage your databases.

   You can also administer and monitor the database with Oracle Enterprise Manager Grid Control, which is installed by default with Oracle Database. Oracle Enterprise Manager Grid Control includes the Oracle Management Agent, Oracle Management Service, and Oracle Management Repository, and Grid Control, a browser-based central console through which administrators can perform all monitoring, administration, and configuration tasks for the enterprise.

   See Also: *Oracle Enterprise Manager Grid Control Installation and Basic Configuration* available on the Enterprise Manager Grid Control installation media

To perform advanced administration tasks, such as monitoring Oracle Database and managing multiple hosts, application servers, and databases including the one that you are installing, install Oracle Enterprise Manager as follows:

1. Use this guide to install Oracle Database.

If you plan to use Oracle Real Application Clusters, then install Oracle Database by using platform-specific *Oracle Clusterware Installation Guide* and *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

2. Use *Oracle Enterprise Manager Grid Control Installation and Basic Configuration* to install and configure Oracle Enterprise Manager. For postconfiguration tasks, use *Oracle Enterprise Manager Advanced Configuration*.

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. To install the Oracle Internet Directory server components, run Oracle Universal Installer from an Oracle 10*g* Application Server installation.

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 Security Administrator's Guide*
- *Oracle Label Security Administrator's Guide*
- *Oracle Application Server Security Guide*
- Oracle Technology Network topics on database security (`http://www.oracle.com/technology/deploy/security/index.html`)

How do I use Oracle Database to manage my XML data?

Use Oracle XML DB, which is installed as part of Oracle Database. Oracle XML DB enables you to efficiently store, generate, retrieve, query, and manage XML data on your site. Oracle XML DB provides all the advantages of a relational database, for example, allowing you to control the referential integrity of XML data with constraints and triggers. It works well with large amounts of XML data by storing it in a parsed, relational form, which improves access performance.

Oracle XML DB supports XML Type, which is a native data type for XML data, for which you can choose various storage options depending on your needs. In addition,

Oracle XML DB supports XML Schema processing, structured and unstructured storage, a content repository that you can access by using common protocols (FTP, HTTP(S), and WebDAV), and SQL/XML, which is a standard for SQL with XML. For Oracle Database 11*g* Release 1 (11.1), Oracle XML DB introduced support for the XQuery language for querying, transforming, and constructing XML; the ability for users to define their own metadata for schema-based XML; a set of new SQL functions for DML operations on XML data; and more.

You can use Oracle XML DB in conjunction with Oracle XML Developer's Kit (XDK) to build applications that run on either Oracle Database or Oracle Application Server.

See Also:

- *Oracle XML DB Developer's Guide*
- *Oracle XML Developer's Kit Programmer's Guide*

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* to install Oracle OLAP:

- When you run Oracle Universal Installer, select the **Custom** installation type, and in the Available Product Components screen, select **Oracle OLAP**.

  See Also:

  - *Oracle OLAP User's Guide*
  - *Oracle OLAP DML Reference*
  - *Oracle OLAP 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 this guide to install Oracle Data Mining:

1. When you run Oracle Universal Installer, select the **Enterprise Edition** installation type.
2. In the Select Database Configuration screen, select the **General Purpose/Transaction Processing** configuration.

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)

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 User's Guide*
- *Oracle Database Backup and Recovery Reference*

Is Oracle Workflow included with Oracle Database 11*g*?

Starting with Oracle Database 11*g*, Oracle Workflow is no longer released with the database. Oracle Workflow will be available with the Oracle E-Business Suite releases.

See Also: Oracle Workflow statement of direction (`http://www.oracle.com/technology/products/ias/workflow/workflow_sod.html`)

Is there a migration plan for customers that have built solutions using Oracle Workflow?

Starting January 2006, customers are encouraged to re-create and implement workflows using Oracle BPEL Process Manager. Oracle is in the process of creating a technical migration guide that will provide detailed recommendations for migrating Oracle Workflow processes to Oracle BPEL Process Manager.

See Also: Oracle Workflow statement of direction (`http://www.oracle.com/technology/products/ias/workflow/workflow_sod.html`)

Installing Oracle Database with Oracle Applications

The following are frequently asked questions with respect to installing Oracle database with Oracle applications:

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.

If you need to implement your applications with Oracle Real Applications Clusters databases, refer to *Oracle Real Application Clusters Installation Guide for Linux and UNIX* and platform-specific *Oracle Clusterware Installation Guide*.

How can I create Web applications that communicate with Oracle Database?

Install Oracle Application Express and a web server:

Use this guide to install Oracle Database. Oracle Application Express is automatically installed, when you install Oracle database.

Which Web server can my Oracle applications use?

Install Oracle HTTP Server:

Use this guide to install Oracle Database.

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`

Installing Oracle Database Heterogeneous Connectivity Tools (Gateways)

The following section discusses about Gateway products:

How can my Oracle applications access data in a non-Oracle database system?

How can my Oracle applications access data in a non-Oracle database system?

You can use Oracle Database Gateway as the connectivity tool to enable Oracle applications to access data in non-Oracle databases. The following are the functions of Oracle Database Gateway:

- Integrates a non-Oracle database into your Oracle Database environment.
- Enables Oracle PL/SQL applications to integrate with APPC-enabled transactions, or access messages in IBM Websphere MQ.

You can install the Gateway product on a computer independent of the Oracle application, Oracle database, and non-Oracle database.

For example, suppose you have the following scenario:

- Oracle Database is installed on an UNIX computer.
- The Oracle application is installed on a Microsoft Windows computer and accesses data from the Oracle database on the UNIX computer.
- The Oracle application must join data in a DB2 database on Solaris Operating System and an Oracle Database on UNIX.

You have the option of installing the Database Gateway for DRDA on the Solaris computer where DB2 is running, on UNIX where Oracle is running, or on a third computer.

Table H–1 lists the non-Oracle database systems that you can access from Oracle applications, and the Gateways products that are available for those systems.

Table H–1 Oracle Gateway Products

Non-Oracle Database	Oracle Gateway Products and Documentation
IBM DB2 Universal Database (UDB)	Oracle Database Gateway for DRDA. Use *Oracle Database Gateway Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64* and *Oracle Database Gateway for DRDA User's Guide*.
IBM DB2 z/OS	Oracle Database Gateway for DRDA. Use *Oracle Database Gateway Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64* and *Oracle Database Gateway for DRDA User's Guide*.
IBM DB2/400	Oracle Database Gateway for DRDA. Use *Oracle Database Gateway Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64* and *Oracle Database Gateway for DRDA User's Guide*.
WebSphere MQ	Oracle Database Gateway for WebSphere MQ. *Oracle Database Gateway for WebSphere MQ Installation and User's Guide*.
CICS/TS IMSTM	Oracle Database Gateway for APPC. Use *Oracle Database Gateway for APPC Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), and Linux x86*.
SQL Server	Oracle Database Gateway for SQL Server. Use *Oracle Database Gateway Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64* and *Oracle Database Gateway for SQL Server User's Guide*.
Sybase Adaptive Server	Oracle Database Gateway for Sybase. Use *Oracle Database Gateway Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64* and *Oracle Database Gateway for Sybase User's Guide*.
Teradata	Oracle Database Gateway for Teradata. Use *Oracle Database Gateway Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64* and *Oracle Database Gateway for Teradata User's Guide*.
Informix Server	Oracle Database Gateway for Informix. Use *Oracle Database Gateway Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64* and *Oracle Database Gateway for Informix User's Guide*.

Table H–1 (Cont.) Oracle Gateway Products

Non-Oracle Database	Oracle Gateway Products and Documentation
IMS	Oracle Database Gateway for IMS. Use *Oracle Database Gateway for IMS, VSAM, and Adabas Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64*, *Oracle Database Gateway for IMS User's Guide* and *Oracle Connect for IMS, VSAM, and Adabas Gateways Installation and Configuration Guide for IBM z/OS*
VSAM	Oracle Database Gateway for VSAM. Use *Oracle Database Gateway for IMS, VSAM, and Adabas Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64*, *Oracle Database Gateway for VSAM User's Guide* and *Oracle Connect for IMS, VSAM, and Adabas Gateways Installation and Configuration Guide for IBM z/OS*.
Adabas	Oracle Database Gateway for Adabas. Use *Oracle Database Gateway for IMS, VSAM, and Adabas Installation and Configuration Guide for AIX 5L Based Systems (64-Bit), HP-UX PA-RISC (64-Bit), Solaris Operating System (SPARC 64-Bit), Linux x86, and Linux x86-64*, *Oracle Database Gateway for Adabas User's Guide* and *Oracle Connect for IMS, VSAM, and Adabas Gateways Installation and Configuration Guide for IBM z/OS*.

Country Codes

This appendix contains a list of valid country codes that can be used while installing Oracle Configuration Manager.

Valid Country Codes

Table I–1 contains a list of countries and their short names (codes.)

Table I–1 Country Codes

Country	Short Name (Code)
African Other	AA
Andorra	AD
United Arab Emirates	AE
Afghanistan	AS
Antigua and Barbuda	AM
Anguilla	AI
Albania	AL
Armenia	AM
Netherlands Antilles	AN
Angola	AO
Antarctica	AQ
Argentina	AR
American Samoa	AS
Austria	AT
Australia	AU
Aruba	AW
Azerbaijan	AZ
Bosnia-Herzegovina	BA
Barbados	BB
Bangladesh	BD
Belgium	BE
Burkina Faso	BF

Table I–1 (Cont.) Country Codes

Country	Short Name (Code)
Bulgaria	BG
Bahrain	BH
Burundi	BI
Benin	BJ
Bermuda	BM
Brunei Darussalam	BN
Bolivia	BO
Brazil	BR
Bahamas	BS
Bhutan	BT
Bouvet Island	BV
Botswana	BW
Belarus	BY
Belize	BZ
Canada	CA
Cocos (Keeling) Islands	CC
Central African Republic	CF
Congo	CG
Switzerland	CH
Cote D'Ivoire	CI
Cook Islands	CK
Chile	CL
Cameroon	CM
China	CN
Columbia	CO
Costa Rica	CR
Cuba	CU
Cape Verde	CV
Christmas Island	CX
Cyprus	CY
Czech Republic	CZ
Germany	DE
Djibouti	DJ
Denmark and Iceland	DK
Dominica	DM
Dominican Republic	DO
Algeria	DZ

Table I–1 (Cont.) Country Codes

Country	Short Name (Code)
Ecuador	EC
Estonia	EE
Egypt	EG
Western Sahara	EH
Eritrea	ER
Spain	ES
Ethiopia	ET
Finland	FI
Fiji	FJ
Falkland Islands (Malvinas)	FK
Micronesia (Federated States Of)	FM
Faroe Islands	FO
France	FR
France - Overseas Territories	FX
Gabon	GA
United Kingdom	GB
Grenada	GD
Georgia	GE
French Guiana	GF
Ghana	GH
Gibraltar	GI
Greenland	GL
Gambia	GM
Guinea	GN
Guadeloupe	GP
Equatorial Guinea	GQ
Greece	GR
South Georgia and South Sandwich Island	GS
Guatemala	GT
Guam	GU
Guinea - Bissau	GW
Guyana	GY
Hong Kong	HK
Heard Island and McDonald Islands	HM
Honduras	HN
Croatia	HR
Haiti	HT

Table I–1 (Cont.) Country Codes

Country	Short Name (Code)
Hungary	HU
Indonesia	ID
Ireland	IE
Israel	IL
India	IN
British Indian Ocean Territory	IO
Iraq	IQ
Iran (Islamic Republic of)	IR
Iceland	IS
Italy	IT
Jamaica	JM
Jordan	JO
Japan	JP
Kenya	KE
Kyrgyzstan	KG
Cambodia	KH
Kiribati	KI
Comoros	KM
Saint Kitts and Nevis	KN
Democratic People's Republic of Korea	KP
Republic of Korea	KR
Kuwait	KW
Cayman Islands	KY
Kazakhstan	KZ
Lao People's Democratic Republic	LA
Lebanon	LB
Saint Lucia	LC
Liechtenstein	LI
Sri Lanka	LK
Liberia	LR
Lesotho	LS
Lithuania	LT
Luxembourg	LU
Latvia	LV
Libyan Arab Jamahiriya	LY
Morocco	MA
Monaco	MC

Table I–1 (Cont.) Country Codes

Country	Short Name (Code)
Republic of Moldova	MD
Madagascar	MG
Marshall Islands	MH
Macedonia	MK
Mali	ML
Myanmar	MM
Mongolia	MM
Macau	MO
Northern Mariana Islands	MP
Martinique	MQ
Mauritania	MR
Montserrat	MS
Malta	MT
Mauritius	MU
Malawi	MW
Mexico	MX
Malyasia	MY
Mozambique	MZ
Namibia	NA
New Caledonia	NC
Niger	NE
Norfolk Island	NF
Nigeria	NG
Nicaragua	NI
Netherlands	NL
Norway	NO
Nepal	NP
Narau	NR
Niue	NU
New Zealand	NZ
Oman	OM
Panama	PA
Peru	PE
French Polynesia	PF
Papua New Guinea	PG
Philippines	PH
Pakistan	PK

Table I–1 (Cont.) Country Codes

Country	Short Name (Code)
Poland	PL
Saint Pierre and Miquelon	PM
Pitcairn	PN
Puerto Rico	PR
Portugal	PT
Palau	PW
Paraguay	PY
Qatar	QA
Reunion	RE
Romania	RO
CIS-Comm. of Indep. States	RU
Rwanda	RW
Saudi Arabia	SA
Solomon Islands	SB
Seychelles	SC
Sudan	SD
Sweden	SE
Singapore	SG
Saint Helena	SH
Slovenia	SI
Svalbard and Jan Mayen Islands	SJ
Slovakia	SK
Sierra Leone	SL
San Marino	SM
Senegal	SN
Somalia	SO
Suriname	SR
Sao Tome and Principe	ST
El Salvador	SV
South Asia Growth Economies	SX
Syrian Arab Republic	SY
Swaziland	SZ
Turks and Caicos Islands	TC
Chad	TD
French Southern Territories	TF
Togo	TG
Thailand	TH

Table I–1 (Cont.) Country Codes

Country	Short Name (Code)
Tajikistan	TJ
Tokelau	TK
Turkmenistan	TM
Tunisia	TN
Tonga	TO
East Timor	TP
Turkey	TR
Trinidad and Tobago	TT
Tuvalu	TV
Taiwan - Republic of China	TW
United Republic of Tanzania	TZ
Ukraine	UA
Uganda	UG
United States Minor Outlying Islands	UM
United States	US
Uruguay	UY
Uzbekistan	UZ
Vatican City State (Holy See)	VA
Saint Vincent and the Grenadines	VC
Venezuala	VE
Virgin Islands (British)	VI
Vietnam	VN
Vanuatu	VU
Wallis and Futuna Islands	WF
Samoa	WS
Yemen	YE
Mayotte	YT
Serbia and Montenegro	YU
South Africa	ZA
Zambia	ZM
Zaire	ZR
Zimbabwe	ZW

Glossary

Automatic Storage Management disk group

A set of disk devices that **Automatic Storage Management** manages as a single unit. Each disk device can be an individual physical disk, a multiple disk device such as a RAID storage array or logical volume, or even a partition on a physical disk. You can create the Automatic Storage Management disk group when you create the **Automatic Storage Management instance**, or with Oracle Database Configuration Assistant.

Automatic Storage Management instance

The Oracle instance that manages Automatic Storage Management disk groups **Automatic Storage Management disk group**s. It is created automatically when you install and configure **Automatic Storage Management**. See also Oracle system identifier (**SID**).

Automatic Storage Management

Enables creation of a single disk group from a collection of individual disk devices. It balances I/O to the disk group across all of the devices in the disk group. It also implements striping and mirroring to improve I/O performance and data reliability.

automatic undo management mode

A mode of Oracle Database in which undo data is stored in a dedicated **undo tablespace**. Unlike in **manual undo management mode**, the only undo management that you must perform is the creation of the undo tablespace. All other undo management is performed automatically.

connect descriptor

A specially formatted description of the destination for a network connection. A connect descriptor contains destination service and network route information.

The destination service is indicated by using its service name for the Oracle Database or its Oracle system identifier (**SID**) for Oracle release 11.1 databases. The network route provides, at a minimum, the location of the **listener** through use of a network address.

connect identifier

A name, net service name, or service name that resolves to a connect descriptor. Users initiate a connect request by passing a user name and password along with a connect identifier in a connect string for the service to which they want to connect, for example:

```
SQL> CONNECT user_name@connect_identifier
Enter password: password
```

control files

Files that record the physical structure of a database and contain the database name, the names and locations of associated databases and online **undo tablespace**, the time stamp of the database creation, the current log sequence number, and checkpoint information.

default domain

The network domain within which most client requests take place. It can be the domain where the client resides, or a domain from which the client often requests network services. The default domain is also the client configuration parameter that determines what domain to append to unqualified network name requests. A name request is unqualified if it does not have a "." character within it.

directory naming

A **naming method** that specifies a directory server to resolve a net service name into a connect descriptor. The net service name is stored centrally in a directory server.

directory server

A Lightweight Directory Access Protocol (LDAP)-compliant directory server. A directory can provide centralized storage and retrieval of database network components, user and corporate policies preferences, user authentication, and security information, replacing client-side and server-side localized files.

external procedures

Procedure or function written in the C programming language and stored in a shared library. An Oracle server can call external procedures or functions using PL/SQL routines. For Oracle Database to connect to external procedures, the server must be configured with a net service name and the **listener** must be configured with protocol address and service information.

global database name

The full database name that uniquely distinguishes it from any other database in your network domain.

For example:

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

where `sales` is the name you want to call your database and `us.example.com` is the network domain in which the database is located.

initialization parameter file

An ASCII text file that contains information needed to initialize a database and **instance**.

instance

Process associated with a running Oracle Database instance. When a database is started on a database server (regardless of the type of computer), Oracle Database allocates a memory area called the **System Global Area** and starts one or more Oracle Database processes. This combination of the System Global Area and Oracle Database processes is called an instance. The memory and processes of an instance manage the associated database's data efficiently and serve the users of the database.

installation type

A predefined component set that automatically selects which components to install. See "Oracle Database Installation Types" on page 1-10 for a list of installation types available with each top-level component.

Interprocess Communication (IPC)

A protocol that client applications use that resides on the same node as the **listener** to communicate with the database. IPC can provide a faster local connection than TCP/IP.

listener

A process that resides on the server and whose responsibility is to listen for incoming client connection requests and manage the traffic to the server.

When a client requests a network session with a database server, a listener receives the actual request. If the client information matches the listener information, then the listener grants a connection to the database server.

listener.ora file

A configuration file for the listener that identifies the:

- Listener name
- Protocol addresses on which it is accepting connection requests
- Services for which it is listening

The `listener.ora` file resides in the `$ORACLE_HOME/network/admin` directory.

An Oracle Database 11*g* Release 1 (11.1) does not require identification of the database service because of service registration. However, static service configuration is required for an Oracle Database 11*g* Release 1 (11.1) if you plan to use Oracle Enterprise Manager.

local naming

A **naming method** that resolves a net service name into a connect descriptor. This name is configured and stored in the **tnsnames.ora file** on each individual client.

manual undo management mode

A mode of the database in which undo blocks are stored in user-managed rollback segments.

naming method

A resolution method used by a client application to resolve a connect identifier to a network address when attempting to connect to a database service. Oracle Net Services supports the following naming methods:

- Local naming
- Directory naming
- Host naming
- External naming

net service name

A simple name for a service that resolves to a connect descriptor. Users initiate a connect request by passing a user name and password along with a net service name in a connect string for the service to which they want to connect:

```
SQL> CONNECT user_name@net_service_name
Enter password: password
```

Depending on your needs, net service names can be stored in a variety of places, including:

- Local configuration file, `tnsnames.ora`, on each client
- Directory server
- External naming service, such as Network Information Service (NIS) or Cell Directory Service (CDS)

OPS$

Acronym for operating system specific. The initialization file parameter `OS_AUTHENT_PREFIX` enables users to specify a prefix that Oracle uses to authenticate users attempting to connect to the database. Oracle concatenates the value of this parameter to the beginning of the user's operating system account name and password. When a connection request is attempted, Oracle compares the prefixed user name with Oracle user names in the database.

The default value of this parameter is `""` (a null string), thereby eliminating the addition of any prefix to operating system account names. In earlier releases, `OPS$` was the default setting.

ORACLE_BASE

`ORACLE_BASE` is the root of the Oracle Database directory tree. The Oracle Base directory is the top level directory that you can use to install the various oracle software products. You can use the same Oracle base directory for more than one installation. For example, `/u01/app/oracle` is an Oracle base directory created by the oracle user.

ORACLE_HOME

Corresponds to the environment in which Oracle Database products run. If you install an OFA-compliant database, using Oracle Universal Installer defaults, Oracle home (known as `$ORACLE_HOME` in this guide) is located beneath `$ORACLE_BASE`. The default Oracle home is `db_`*n* where *n* is the Oracle home number. It contains subdirectories for Oracle Database software executables and network files. See also **Oracle home**.

Oracle home

The directory path in which to install Oracle components (for example, `/u01/app/oracle/product/11.1.0/db_`*n*). You are prompted to enter an Oracle home in the Path field of the Specify File Locations window. See also **ORACLE_HOME**, **Oracle home name**.

Oracle home name

The name of the current Oracle home, for example, `Db_1`. Each Oracle home has a home name that distinguishes it from all other Oracle homes on your computer. During installation, you are prompted to enter an Oracle home name in the Name field on the Specify File Locations window.

Oracle schema

A set of rules that determine what can be stored in an LDAP-compliant directory server. Oracle has its own schema that is applied to many types of Oracle entries,

including Oracle Net Services entries. The Oracle schema for Oracle Net Services entries includes the attributes the entries may contain.

Oracle Documentation Library

The media in your kit that includes the Oracle Database documentation. The Oracle Documentation Library is separate from the installation media.

The Oracle Documentation Library does not include this installation guide or *Oracle Database Release Notes for Linux*. These documents are included on the media labeled Oracle Database 11*g* Release 1 (11.1) and are available on Oracle Technology Network (OTN).

Oracle Net foundation layer

A networking communication layer that establishes and maintains the connection between the client application and server, and exchanging messages between them.

protocol address

An address that identifies the network address of a network object.

When a connection is made, the client and the receiver of the request, such as the **listener**, or Oracle Connection Manager, are configured with identical protocol addresses. The client uses this address to send the connection request to a particular network object location, and the recipient "listens" for requests on this address. It is important to install the same protocols for the client and the connection recipient, and to configure the same addresses.

raw partitions

Portions of a physical disk that are accessed at the lowest possible disk (block) level.

redo log files

Files that contain a record of all changes made to data in the database buffer cache. If an instance failure occurs, then an administrator can use the redo log files to recover the modified data that was in memory.

repository

A set of tables located in any Oracle database accessible to the Oracle Management Server. Oracle Management Server uses a repository to store all system data and application data, information about the state of managed nodes distributed throughout the environment, and information about the separately licensable management packs.

service registration

A feature by which the PMON process (an instance background process) automatically registers information with a **listener**. Because this information is registered with the listener, the **listener.ora file** does not need to be configured with this static information.

Service registration provides the listener with the following information:

- Service name(s) for each running instance of the database
- Instance name(s) of the database
- Service handlers (dispatchers and dedicated servers) available for each instance

  This allows the listener to direct a client's request appropriately.
- Dispatcher, instance, and node load information

This allows the listener to determine which dispatcher can best handle a client connection's request. If all dispatchers are blocked, the listener can spawn a dedicated server for the connection.

This information allows the listener to determine how best to service a client connection request.

SID

The Oracle system identifier that distinguishes the database from all other databases on your computer. The SID automatically defaults to the database name portion of the global database name (`sales` in the example `sales.us.example.com`) until you reach eight characters or enter a period. You can accept or change the default value.

The SID can also refer to an Automatic Storage Management instance SID, available when you install **Automatic Storage Management**.

sqlnet.ora file

A configuration file for the client or server that specifies the:

- Client domain to append to unqualified service names or net service names
- Order of naming methods for the client to use when resolving a name
- Logging and tracing features to use
- Route of connections
- External naming parameters
- Oracle Advanced Security parameters

The `sqlnet.ora` file resides in `$ORACLE_HOME/network/admin`.

Secure Sockets Layer (SSL)

An industry standard protocol designed by Netscape Communications Corporation for securing network connections. SSL provides authentication, encryption, and data integrity using public key infrastructure (PKI).

SSL

See **Secure Sockets Layer (SSL)**.

System Global Area

A group of shared memory structures that contain data and control information for an Oracle Database **instance**.

system identifier

See **SID**.

tablespace

A logical storage unit within a database. Tablespaces are divided into logical units of storage called segments, which are further divided into extents.

tnsnames.ora file

A configuration file that contains net service names mapped to connect descriptors. This file is used for the local naming method. The `tnsnames.ora` file resides in `$ORACLE_BASE/network/admin`.

undo tablespace

An tablespace that contains one or more undo segments. The creation of any other types of segment (for example, tables, indexes) in undo tablespaces is not allowed.

In the automatic mode, each Oracle instance is assigned one and only one undo tablespace. Each undo tablespace is composed of a set of undo files. Undo blocks are grouped in extents. At any point in time, an extent is either allocated to (and used by) a transaction table, or is free.

Blocks in undo tablespaces are grouped into the following categories:

- File control blocks, bitmap blocks, and so forth used for space management
- Undo segments containing transaction table blocks, undo blocks, and extent-map blocks used for transaction management
- Free blocks that are unallocated to file control or undo segments

unqualified name

A net service name that does not contain a network domain.

Index

Symbols

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X