Oracle® Database

Installation Guide

11*g* Release 2 (11.2) for HP OpenVMS Itanium

E56668-01

June 2015

ORACLE®

Oracle Database Installation Guide, 11*g* Release 2 (11.2) for HP OpenVMS Itanium

E56668-01

Primary Author: Nisha Sridhar

Contributors: Dave Hayter, Gary Huffman, Marc Noel, Chris Schuetz, David Miller, Kevin Duffy, Steve Holck, Grant Hayden, Gary Tate

Contents

2 Preinstallation Tasks

3 Installation Tasks

4 Postinstallation Tasks

5 Getting Started with Oracle Database

Preface

This guide and *Oracle Database Administrator's Reference for HP OpenVMS Itanium* provides instructions for installing and configuring Oracle Database 11*g* Release 2 (11.2) on HP OpenVMS Itanium systems.

Documentation for the products that are included with this software is available in the Oracle Database generic documentation set.

Intended Audience

This document is intended for anyone responsible for installing Oracle Database 11*g* Release 2 (11.2) on HP OpenVMS Itanium systems.

Documentation Accessibility

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

Access to Oracle Support

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

Related Documentation

Information about system administration and tuning for a production database system is provided in the following documents:

- *Oracle Database Administrator's Reference for HP OpenVMS Itanium*
- *Oracle Database Administrator's Guide*
- *Oracle Database Net Services Administrator's Guide*
- *Oracle Database Net Services Reference*
- *Oracle Database Performance Tuning Guide*

Information about upgrading from a previous version of Oracle Database is provided in *Oracle Database Upgrade Guide*.

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

- *Oracle Database Installation Guide for HP OpenVMS Itanium*
- *Oracle Database Release Notes for HP OpenVMS Itanium*

You can review these documents at

http://docs.oracle.com/en/database/database.html

Conventions

The following text conventions are used in this document:

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

1 Installation Overview

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

- Overview of Oracle Database Installation
- Changes and Enhancements
- Oracle Database Installation Types
- Overview of Database Configuration Assistant
- Database Configuration Options
- Available Installation Options
- Installation Considerations
- Important Upgrade and Compatibility Issues
- Upgrade Issues
- Cluster File System

1.1 Overview of Oracle Database Installation

The Oracle Database installation process consists of five phases:

1. **Read the release notes:** Read *Oracle Database Release Notes for HP OpenVMS Itanium* before you begin the installation. The release notes are available with the platform-specific documentation. The latest version of the release notes is available at

   `http://docs.oracle.com/en/database/database.html`

2. **Planning the installation:** This chapter describes the Oracle products that you can install and issues that you must consider before starting the installation.
3. **Completing preinstallation tasks:** Chapter 2 describes preinstallation tasks that you must complete before installing the product.
4. **Installing software:**

   - Chapter 3 describes how to use Oracle Universal Installer to install Oracle Database.

     If you want to install client tools, applications, and client interfaces that are not included on the Oracle Database DVD, then see the documentation for those products.

- Chapter 6 describes how to remove Oracle Database.
- Appendix B provides information on performing noninteractive (silent) installations, which you may want to use if you must perform multiple installations of Oracle Database.
- Appendix C describes how to install, configure, and run the Oracle Enterprise Manager Agent.
- Appendix F provides troubleshooting advice in case you encounter problems with the installation.
- Appendix G describes how to install and configure Apache Server.

5. **Completing postinstallation tasks:**
   - Chapter 4 describes recommended and required postinstallation tasks.
6. **Get started using Oracle Database:** Chapter 5 describes how to check the contents of the installed Oracle Database, how to start various tools, and how to locate various files. Appendix D explains how to manage Oracle Database port numbers.

1.2 Changes and Enhancements

The following changes and enhancements are unique to this release of Oracle Database:

- Turning Off the Image Dump
- Oracle Database Dependency on ODS-5 Disk Structure
- Oracle Database Installation Directory Structure

1.2.1 Turning Off the Image Dump

In earlier releases, when one of the background processes terminated abruptly with an access violation, no useful information was readily available regarding the cause for termination. For troubleshooting, certain actions needed to be taken, and the failure needed to be reproduced. Starting with this release, by default, an image dump is created when a background process terminates abruptly. If you do not want an Oracle background process to generate image dump files, then create an HP OpenVMS system logical for that background process as follows:

```
$ DEFINE/SYSTEM ORA_sid_bgprocess_DMP FALSE
```

In the preceding command, the value of *bgprocess* can be PMON, SMON, DBW0, and so on. The value of *sid* should be set to the required system Identifier or instance name.

Start the instance and open the database. For example:

> **Note:** HP OpenVMS logicals must be defined before starting the instance.

```
$ DEFINE/SYSTEM ORA_PROD_LMD0_DMP FALSE
$ SQLPLUS/NOLOG
SQL> CONNECT SYS / AS SYSDBA
Enter password:
SQL> STARTUP
```

This change has no effect on the usual daily production activities. Oracle recommends that you do not define the `dump` logical.

1.2.2 Oracle Database Dependency on ODS-5 Disk Structure

On-Disk Structure Level 5 (ODS-5) format enables HP OpenVMS to support Linux and Unix style file name formats. Longer file names, additional character support, and the ability to have lowercase and mixed-case file names are some of the features supported on ODS-5 disks.

Beginning with OpenVMS 11*g*R2 Oracle Database, it is required to use the ODS-5 format for all disks involved with the installation. This includes the login, download, install target, temporary install, database, and archive/redo logs devices.

> **Note:** Oracle recommends that you read *HP OpenVMS Guide to Extended File Specifications* before changing the structure of any disk. This guide describes ODS-2 and ODS-5 formats in more detail and provides important information that you must follow.

The disks on which the Oracle code tree resides must be formatted in Files-11 ODS-5 format. This is the disk to which the HP OpenVMS logicals `ORA_ROOT`, `ORA_ROOTDIR`, and `ORACLE_HOME` point. The directory that is referenced by these logicals is the parent directory for all Oracle Database software, such as Apache, CTX, INSTALL, NETCONFIG, RDBMS, and UTIL. With HP OpenVMS 11*g* Release 2 (11.2), the only format that will perform as expected is ODS-5.

> **See Also:** Oracle Note 2012766.1 on My Oracle Support.
>
> Alternatively, use the search function and search for ODS-5 for more information.

Precautions

You must observe the following precautions:

- Do not change the format of any HP OpenVMS disk without first making a backup of the disk.
- Conversion from ODS-2 to ODS-5 does not require you to restore files.
- Conversion from ODS-5 to ODS-2 requires you to restore files.

> **Note:** Contact Hewlett-Packard for more information about reverting to ODS-2 format from ODS-5 format.

Checking Disk Format

To check the current disk format, run the following DCL command:

```
$ SHOW DEVICE/FULL device_name
```

The `Volume Status:` line of the output displays the current ODS level. The default when initializing disks is ODS-2.

Changing Disk Format

If you plan to reformat an existing (in use) disk from ODS-2 to ODS-5 format, then:

1. Create at least one HP OpenVMS backup of the disk.

2. Use the following command:

```
$ SET VOLUME /STRUCTURE_LEVEL=5 device_name
```

Example:

```
$ SET VOLUME/STRUCTURE_LEVEL=5  $2$DKC0:
```

Formatting a New Disk

If you plan to use a new (empty) disk, then use the following command to format the new disk to ODS-5 format:

```
$ INITIALIZE/STRUCTURE=5 device_name: volume_label
```

Example:

```
INITIALIZE/STRUCTURE=5 $2$DKC100: TESTVOL
```

1.2.3 Oracle Database Installation Directory Structure

The installation directory structure of Oracle Database closely resembles that of the installation directory structure of Oracle Database on UNIX systems. By making the HP OpenVMS installation directory structure similar to that of other platforms, a more consistent and predictable layout is followed across platforms. This enables database administrators (DBAs) from other platforms to quickly become familiar with Oracle Database for this platform.

1.3 Oracle Database Installation Types

You can choose one of the following installation types when installing Oracle Database:

- **Enterprise Edition:** Installs licensable Oracle Database options and database configuration and management tools. It also installs products most commonly used for data warehousing and transaction processing.
- **Standard Edition:** Designed for department-level or workgroup-level applications and for small-sized or medium-sized enterprises. It provides core relational database options and management services.
- **Custom:** Enables you to select the individual components to install from a list of available components.

Note: ■Oracle Database Client is installed in a separate Oracle home. Do not install Oracle Database Client in the Oracle home used for Oracle Database software.

- Oracle Enterprise Manager 11*g* Release 2 (11.2) Agent is available in a separate Enterprise Manager Grid Control release.

See Also:

- *Oracle Enterprise Manager Concepts* and *Oracle Enterprise Manager Grid Control Installation and Basic Configuration* for more information about Oracle Enterprise Manager 11*g* Release 2 (11.2).
- *Oracle Database Licensing Information* guide for more information about the features available with each Oracle Database edition and information about licensing.

1.4 Overview of Database Configuration Assistant

Database Configuration Assistant (DBCA) enables you to copy a preconfigured Oracle Database, or to create a fully customized database to match the selected environment and database configuration. In order to run DBCA, type DBCA at the DCL prompt.

When installing Oracle Database software using any database configuration option other than Custom and Software Only, Oracle Universal Installer prompts for a global database name and system identifier (SID). After Oracle Database installation is complete, Database Configuration Assistant uses this information to create a database.

The following list describes database configurations:

- General Purpose

  If you select this option with an Enterprise Edition installation, then Database Configuration Assistant creates a preconfigured, ready-to-use, multipurpose database with the following features:

  – Default initialization parameters using server parameter file and Automatic Undo Management.

  – Automatic installation and configuration of Oracle Options and *inter*Media.

  – Advanced Replication capabilities.

  – Database configuration of dedicated server mode.

  – Archiving mode of `NOARCHIVELOG`.

- Transaction Processing

  If you select this option, then Database Configuration Assistant creates a preconfigured, ready-to-use OLTP database.

- Data Warehouse

  If you select this option, then Database Configuration Assistant creates a database that is fully enabled for data warehousing applications.

- Customized

  If you select this option, then Database Configuration Assistant guides you in creating a fully customized database.

- Automatic or Manual

  Automatic or manual configuration options, Oracle Text components, and advanced replication.

  Select this option only if you are experienced with advanced database creation procedures. You must customize some or all of the following settings or parameters:

  - Data, control, and redo log file settings.
  - Tablespace and extent sizes.
  - Database memory parameters.
  - Archiving modes, formats, and destinations.
  - Trace file destinations.
  - Character set values.

- For HP OpenVMS 11*g* Release 2 (11.2), "*Software Only*" is the method of install. Once the install is complete, DBCA can be run from the command line to create a database.

  See Also:

  - Section B.5, "Running Database Configuration Assistant in Noninteractive or Silent Mode" for more information about running Database Configuration Assistant from the command line.
  - *Oracle Database Globalization Support Guide* for more information about database character sets.

1.5 Database Configuration Options

During the installation, you can choose whether you want to create an Oracle Database as part of the installation. The Select Configuration Option screen provides the following options:

- Create a database
- Install database Software only

```
Specify ASM SYS Password: [        ]
Confirm ASM SYS Password: [        ]
```

- Install database Software only

If you decide not to create an Oracle Database during the installation, then you can use the Database Configuration Assistant (DBCA) to create one after you have installed the software. Choose the third option from this screen as ASM is not supported on OpenVMS platforms.

See Also: *Oracle Database 2 Day DBA* for more information about using Database Configuration Assistant (DBCA) to create a database after installation

---

If you choose to create an Oracle Database, Oracle Universal Installer uses the Database Configuration Assistant to create it. You can choose to create one of the preconfigured database types, which are designed for different types of applications, modify one of the preconfigured database types, or create a customized database to suit your requirements.

This section describes the following database configuration options:

- Database Options
- Installation Choices that Affect Database Creation

- Creating a Database After Installation

1.5.1 Database Options

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

- General Purpose
- Transaction Process
- Database Warehouse
- Advanced

1.5.1.1 General Purpose

A starter database designed for general purpose usage. Users of this database type perform a variety of database tasks, ranging from simple transactions to complex queries. Select this database environment for general purpose use.

1.5.1.2 Transaction Process

A starter database optimized for transaction-heavy applications. Users of this database type perform large numbers of concurrent transactions, in which each transaction is a relatively simple operation that is processing a small amount of data.

Billing databases, such as those commonly found on Internet commerce websites, are the most common example of this database configuration. These databases are also known as **Online Transaction Processing** (OLTP) databases.

1.5.1.3 Database Warehouse

A starter database optimized for data warehousing applications. Users of this database type typically perform reporting and data analysis tasks, often on data uploaded from operational systems and databases.

1.5.1.4 Advanced

This option allows you to customize the configuration of your starter database. This option enables you to create a customized database configuration that meets specialized requirements. Select this option only if you are prepared to provide detailed product and database environment choices. Selecting this option requires a longer installation session than choosing a preconfigured database.

Oracle recommends that you install at least one preconfigured database to serve as a template for database configuration.

1.5.2 Installation Choices that Affect Database Creation

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

- Noninteractive mode

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

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, then Oracle Universal Installer does not prompt you for database information. Instead, it installs the software and then runs Database Configuration Assistant in interactive mode. Using the screens in Database Configuration Assistant, you can either modify one of the preconfigured database types or create a custom database and specify how you want to configure it.

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

1.5.3 Creating a Database After Installation

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

See Also: *Oracle Database 2 Day DBA* for more information about using Database Configuration Assistant to create a database after installation

1.6 Available Installation Options

The current release of Oracle Database provides new options for simplifying database administration tasks. These include:

- Database Storage Options
- Database Backup and Recovery Options

1.6.1 Database Storage Options

If you choose to create a database during the installation, then you can specify the File System option as the storage option for database files.

File System

Oracle Automatic Storage Management (Oracle ASM) is not supported as a storage option for HP OpenVMS. If you choose the File System option, then Database Configuration Assistant creates the database files in a directory on a device mounted on the system. Oracle recommends that the device that you select be separate from the devices that the operating system or by Oracle products use. The device that you select can be any of the following:

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

If you are using multiple disks in an logical volume manager (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 device for database storage.

See Also: *Oracle Database Administrator's Guide*

1.6.2 Database Backup and Recovery Options

If you choose to use Oracle Enterprise Manager Grid Control during the installation, then you can enable automated database backups that use the default backup strategy recommended by Oracle.

Note: You do not have to enable automated backups during the installation. If you prefer, you can use Oracle Enterprise Manager Grid Control to configure automated backups after you install the software and create a database.

This section contains the following sections:

- Enabling Automated Backups
- Backup Job Default Settings

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 the database files to a storage area on disk 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 have previously installed the Oracle Enterprise Manage Grid Control Agent. During database installation you must specify the following information:

- The location of the flash recovery area.

  You can choose a device directory or accept the default for the flash recovery area. The default disk quota configured for the flash recovery area is 2 GB.

- 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 have the appropriate privileges and rights required for database administrators.

  See Also: Chapter 2 for information about choosing the location of the flash recovery area and determining the disk space requirements and about the requirements for the database administrator account

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 Database Configuration Assistant, either during or after the installation, you can specify a start time for the backup job and a disk quota for the flash recovery area.

See Also:

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

1.7 Installation Considerations

This section contains information about the following considerations, which you must address before deciding how to install Oracle Database:

- Hardware and Software Certification
- Multiple Oracle Homes
- Logical Names and Symbols

1.7.1 Hardware and Software Certification

The hardware and software requirements included in this installation guide are current at the time of publishing this guide. However, because new platforms and operating system software versions may be certified after this guide is published, review the certification matrix on My Oracle Support for an updated list of certified hardware platforms and operating system versions. You can visit My Oracle Support at

`https://support.oracle.com`

1.7.2 Multiple Oracle Homes

This product supports multiple Oracle homes. You can install this release or earlier releases of the software more than once on the same system. For example, you may want to make an Enterprise Edition installation and a Standard Edition installation available on the same system.

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 the Oracle home directory of a different release. For example, you cannot install Oracle Database 11*g* Release 2 (11.2) 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.

1.7.3 Logical Names and Symbols

Note: Before installing Oracle software, you must log out of any existing Oracle sessions. If you do not do this, then the new installation includes the logicals and symbols of the active Oracle sessions.

Ensure that the `LOGIN.COM` file of the account that Oracle Database uses to install does not define any of the Oracle logicals or symbols, and does not run any command file that may define them. Ensure that none of the general Oracle specific logicals (typically beginning with `ORA_`) are defined in the system table, except for some logicals related to mailbox devices and shared libraries. Oracle Database may not run correctly if these logicals are defined.

1.8 Important Upgrade and Compatibility Issues

Note: If you install Oracle Database 11*g* Release 2 (11.2) on a node where other versions of Oracle Database are run, then ensure that you do not have Oracle symbols or logicals defined before installing Oracle Database 11*g* Release 2 (11.2). Do not run any `orauser` scripts as part of the login sequence. Failure to heed this warning will result in numerous problems, including undefined symbols and overwriting of the previous code tree.

Take the following issues into consideration when installing 11.2.0.4 on a system with existing Oracle software:

- Oracle Database 11*g* Release 2 (11.2) Enterprise Edition must be installed in a location separate from other previous Oracle Database installations.
- Oracle products from earlier releases of Oracle Database, such as Release 10.2.0.x and Release 9.2.0.x, must not be linked with an Oracle Database 11*g* Release 2 (11.2) installation.

1.9 Upgrade Issues

For more information about upgrading a previous release of Oracle Database to Oracle Database 11*g* Release 2 (11.2), see *Oracle Database Upgrade Guide*. The following sections includes additional upgrade information that you should review before upgrading an existing database.

1.9.1 Manually Upgrade a Database

To manually upgrade an existing database to Oracle Database 11*g* Release 2 (11.2) on HP OpenVMS, follow the steps in *Oracle Database Upgrade Guide*. In Chapter 3 of the guide, use the following steps for HP OpenVMS.

In the "Prepare the New Oracle Home" subsection of the "Upgrade the Database Manually" section, substitute `ORACLE_HOME/dbs` with `ORA_ROOT:[DBS]`. Also substitute `ORACLE_HOME/dbs/orapw`*sid* with `ORA_ROOT:[DBS]orapw`*sid*.

- In step 1 of the "Upgrade the Database" subsection, ignore the steps for the Windows operating system.

- Set your default directory to the new Oracle home and execute the script `@ORAUSER.COM` (without arguments). Then `SET DEF ORA_ROOT:[RDBMS]` and execute the migrate script: To upgrade 10.2 to 11.2, enter `@MIGRATE102.COM`
- Enter the details prompted by the upgrade assistant.
- Proceed to step 4 in the "Prepare the New Oracle Home" subsection, which involves changing to the ORA_ROOT:[RDBMS.ADMIN] directory.

If you want to use an existing non-VMS database on an HP OpenVMS Itanium system, you must upgrade the non-VMS database to 11*g* Release 2 (11.2) before migrating it to HP OpenVMS Itanium.

1.9.2 AL24UTFFSS Character Set

To upgrade an existing database that uses the `AL24UTFFSS` character set, upgrade the database character set to `UTF8` before upgrading to Oracle Database 11*g* Release 2 (11.2). 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.

1.10 Cluster File System

In this document, there are references to the term **cluster file system**. This term refers to the configuration in which all disks are equally accessible from all nodes on the cluster. This is the most commonly used configuration on an HP OpenVMS cluster. All references to this term should be considered as references to this configuration on an HP OpenVMS cluster.

2

Preinstallation Tasks

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

2.1 Checking Hardware Requirements

> **Note:** Oracle Database 11*g* Release 2 (11.2) must be installed on an ODS-5 formatted disk. See Section 1.2, "Changes and Enhancements" for more information.

The following sections list the minimum hardware requirements for installing Oracle Database products on an HP OpenVMS system:

2.1.1 Minimum Hardware Requirements

The following are the minimum hardware requirements for installing Oracle Database 11*g* Release 2 (11.2):

RAM

A minimum of 2 GB RAM is required to install Oracle Database products. However, Oracle Corporation suggests having at least 4 GB RAM. Oracle Database Client products require 1 GB RAM. Also, consider the associated applications used on your system when calculating RAM.

To determine the amount of RAM installed on the system and the amount of paging currently configured on the system, enter the following command:

```
$ SHOW MEMORY
```

To review memory reservations on the node, run the following command:

```
$ SHOW MEMORY/RESERVED
System Memory Resources on 13-JUL-2002 09:57:11.72

Memory Reservations (pages):   Group   Reserved  In Use  Type
ORA_PROD_SGA                   SYSGBL  5120      0       Allocated
ORA_PROD_SGA                   SYSGBL  5         0       Page Table
Total (40.04 Mbytes reserved)          5125      0
```

See Also:

- Section 2.5.1, "Reserved Memory Registry" for more information.
- Hewlett-Packard documentation for more information about reserving memory.

Chip Set

All HP OpenVMS Itanium chip sets are supported.

DVD Drive

This product is available for download and an optional DVD disk is provided upon request. Please ensure that you have a DVD drive that is supported by HP OpenVMS. Oracle uses ODS-5 format installation media.

2.1.2 Images

Oracle Database Enterprise Edition is installed with three default images, namely, `LIBCLNTSH.SO`, `ORACLE.EXE`, and `ORA_JAVA_VMS_SHR.EXE`.

`LIBCLNTSH.SO` is linked with Oracle Database Client images, reduces their size by removing direct references to Oracle Net Services and other common routines. `ORACLE.EXE` is the Oracle server image and will not be present in a Client-only installation. `ORA_JAVA_VMS_SHR.EXE` is the Oracle Java VM shared image.

In addition, HP OpenVMS 11*g* Release 2 (11.2) has the following four shareable images used by the server: LIBCORENLS11.SO, LIBORASHR11.SO, LIBSKGXN2.SO, and LIBSKGXP11.SO.

2.1.3 Disk Space Requirements

Oracle Universal Installer enables you to select an installation category and type. Your choices determine how much disk space is required. The disk space requirements do not include the size of the database. A production Oracle Database that supports many users requires significant disk space and memory.

Table 2–1 Disk Space Requirements for Oracle Database Server

Installation Type	Required Disk Space
Enterprise Edition	10 GB
Standard Edition	10 GB
Custom	Depends on the components selected, but a minimum of 5 GB
Staging area (copying dvd contents to disk)	5 GB

2.1.4 Oracle Advanced Security

Oracle Advanced Security is an add-on product that is added to the standard Oracle Net Services Server or Oracle Net Services Client. If you purchase it, then install it on both the server and the client.

The SSL version compatibility is SSL3.0 or later.Kerberos and Secure Sockets Layer (SSL) have requirements for authentication protocols that are supported by Oracle Advanced Security. No additional authentication protocol software is required to relink Oracle products. The requirements for Kerberos and SSL are as follows:

- Kerberos

  Requires MIT Kerberos version 5 Release 1.1. The Kerberos authentication server must be installed on a physically secure system.

- SSL

  Requires a wallet that is compatible with the Oracle Wallet Manager version 11.2. Wallets that were created by using earlier releases of the Oracle Wallet Manager are not upward-compatible. Oracle Advanced Security provides and installs SSL.

See Also: For more information about Oracle Advanced Security and system management products, see *Oracle Database Advanced Security Administrator's Guide*

Note: No additional authentication protocol software is required to relink Oracle products. However, Oracle does not provide third-party authentication servers (for example, Kerberos). The appropriate authentication server for these protocols must be installed and configured separately.

SSL is provided and is always installed with Oracle Advanced Security.

2.2 Checking Software Requirements

This section provides information about checking the software requirements.

Checking for Required Software

Depending on the products that you intend to install, verify that the following software is installed on the system. The procedure following the table describes how to check these requirements.

Item	Requirement
Operating system	HP OpenVMS Itanium versions 8.4 and 8.4-1H1
Oracle Universal Installer	X Window and X/Motif software:
Oracle Messaging Gateways	Download the SupportPac MA88 from http://www-947.ibm.com/support/entry/portal/Overview/
Pro*C/C++, Oracle XML Developer's Kit (XDK)	■ HP C 7.3-289 ■ HP C++ 7.4-004
Oracle Net Services Requirements	Oracle Net Services on HP OpenVMS is developed and certified using TCP/IP Services for HP OpenVMS (UCX). If you want to use the TCP/IP protocol adapter for Oracle Net Services, then install version 5.7 ECO 4 or later of TCP/IP Services on HP OpenVMS Itanium. **Caution:** TCP/IP protocol stacks from other vendors may work with Oracle, but this is not guaranteed by Oracle. Any TCP/IP problems that cannot be reproduced using TCP/IP Services for HP OpenVMS should be referred to the TCP/IP vendor. Vendor-provided protocol services are usually upward-compatible, so existing applications continue to work without modification. Therefore, later releases of TCP/IP are upward-compatible with Oracle Net Services, if the vendor-specified application programming interface (API) does not change with new releases.
Oracle JDBC/OCI Drivers	You can use HP JDK 1.5.0 version with the Oracle JDBC/OCI drivers. However, they are not required for the installation. **Note:** HP JDK version 1.5.0 is installed as part of a standard Oracle software installation. HP JDK version 1.6.0 is supported if it is installed with the operating system.
Pro*COBOL	HP COBOL 3.0-0002
Pro*FORTRAN	HP FORTRAN 8.2-104939-50H96

To ensure that the system meets these requirements:

1. To determine which version of HP OpenVMS is installed, enter the following command:

   ```
   $ WRITE SYS$OUTPUT F$GETSYI("VERSION")
   V8.4
   ```

   If necessary, see the operating system documentation for information about upgrading the operating system.

2. To display additional information about the operating system, enter the following command:

   ```
   $ SHOW SYSTEM /NOPROCESS /FULL
   ```

To view information about all nodes in a cluster, add the /CLUSTER qualifier to this command.

3. If you intend to use Oracle Messaging Gateway and require MQSeries classes for Java and MQSeries classes for Java Message Service (SupportPac MA88), download it from the following IBM Web page:

   http://www-947.ibm.com/support/entry/portal/Overview/

4. If you require a Corrective Services Deliverable (CSD) for MQSeries, refer to the following website for download and installation information:

   http://www-01.ibm.com/software/integration/wmq/#

5. To determine the version of TCP/IP installed, enter the following command:

```
$ tcpip show version
HP TCP/IP Services for OpenVMS Industry Standard 64
Version V5.7 - ECO 4
on an HP BL860c i2 (1.33GHz/4.0MB) running OpenVMS V8.4
```

Oracle Net Services HP OpenVMS Mailbox Driver

The Oracle Net Services HP OpenVMS Mailbox driver (protocol IPC) is included in NETWORK. You do not need an Oracle Net Services license to use the HP OpenVMS Mailbox driver.

Checking for Required Patches

You must ensure that the patches described in the following table are installed on the system.

Installation Type or Product	Minimum Requirement
All installations	The following patches, or their later versions, are required:
All installations	■ VMS84I_UPDATE V8.0 ■ VMS84I_FIBRE_SCSI V5.0 ■ VMS84I_IPC V1.0 ■ VMS84I_LAN V3.0 ■ VMS84I_PCSI V4.0 ■ VMS84I_RMS V4
IBM Messaging Gateway	IBM AXPVMS MQ CLIENT V5.30
Oracle Messaging Gateway	Corrective service diskette (CSD) for MQSeries: CSD09 or later for MQSeries V5.3

Note: In some cases, later patch sets would cumulatively include earlier patches and therefore, the preceding command would not explicitly indicate the required patch sets. In such cases, check the Patch Distribution website of Hewlett-Packard for the contents of the patches that are displayed to verify if the required patches are included or not.

To check if you have the required patches installed on the system, enter the following command:

```
$ PRODUCT SHOW HISTORY
```

The output of this command is a list of all software patches or products (or both) that are installed on the system. Verify that you meet the minimum requirements to install and run Oracle Database.

In addition to this document, see My Oracle Support Document 2000927.1 and Index of install related notes for Oracle Database 11*g* Release 2 (11.2) on HP OpenVMS, for an updated list of the following:

- Required patches
- ECO level check scripts
- Preinstallation check scripts
- Quota check scripts

My Oracle Support is available at

https://support.oracle.com

2.3 Installation-Specific Issues and Restrictions

The following factors can affect the installation or use of Oracle Database:

- Section 2.3.1, "New Oracle Home"
- Section 2.3.2, "JDK"
- Section 2.3.3, "Devices"

2.3.1 New Oracle Home

Do not install Oracle Database into the root directory of an existing Oracle installation containing any Oracle Software. Oracle recommends that you install Oracle Database products into a new Oracle Home.

Logical Names

Oracle Database 11*g* Release 2 (11.2) has two separate logical names that specify the top-level directory of the installation. `ORA_ROOT` is a concealed logical name used in earlier releases, and `ORACLE_HOME` is a nonconcealed logical name. You can use both logical names with the following restriction:

If the top-level directory is `DISK$DISK1:[ORACLE11204]`, then the logical names are defined as follows:

```
$ SHOW LOGICAL ORA_ROOT
DISK$DISK1:[ORACLE11204.]
```

```
$ SHOW LOGICAL ORACLE_HOME
DISK$DISK1:[ORACLE11204.]
```

Therefore, the following command is invalid:

```
$ SET DEFAULT ORA_ROOT
```

The following commands are valid:

```
$ SET DEFAULT ORA_ROOT:[000000]
$ SET DEFAULT ORACLE_HOME
```

2.3.2 JDK

The JDK release that is shipped with Oracle Database is used by Oracle Java applications such as Oracle Universal Installer and is the only JDK that is supported to run with these applications. You must not modify this JDK, unless it is done through a patch provided by Oracle Support Services.

2.3.3 Devices

With Oracle 11*g* Release 2 (11.2), all file systems must reside on ODS-5 (On-Disk Structure Level 5) formatted disks. This includes the installation location, login directory, database files, and redo log files. This is required for mixed case file name support.

Oracle Database server must be able to verify that files have been written to disk. Devices that do not support this verification are not supported for use with Oracle Databases, although Oracle software can be installed on them. Datafiles always must reside on volumes with write through cache enabled.

Note: Because of this requirement for verification, third-party software that creates one or more virtual disks in memory must not be used for data files.

2.4 Creating a User Account

This section provides installation procedure requirements. It discusses the following topics:

- Section 2.4.1, "Creating an Oracle Database DBA Account"
- Section 2.4.2, "Adding a Record in the User Authorization File"
- Section 2.4.3, "Setting Account Privileges"
- Section 2.4.4, "Setting Account Quotas"
- Section 2.4.5, "Process Rights for Database Administrators"
- Section 2.4.6, "Process Rights for SGA Protection"
- Section 2.4.7, "Security Issues with Multiple Databases"
- Section 2.4.8, "Completing Account Setup"
- Section 2.4.9, "Verifying Privileges"
- Section 2.4.10, "Setting Permissions for File Creation"

2.4.1 Creating an Oracle Database DBA Account

Create an HP OpenVMS user account to administer and maintain the Oracle installation. Setting up an Oracle Database account is the same as setting up any other HP OpenVMS user account.

To set up an Oracle Database account:

1. Determine the values that you want to enter to create the account.
2. Run the AUTHORIZE utility to add a record in the User Authorization File (UAF).
3. Use AUTHORIZE to set the following account requirements:
   - Account quotas
   - Account privileges
   - Process rights identifiers
4. Exit AUTHORIZE to complete the account setup.

> **See Also:** The Hewlett-Packard documentation for more information about running the AUTHORIZE utility

Determining Account Information

To add an account, you must enter the following information:

- User name
- User password
- USER and GROUP numbers for the UIC (octal values)
- Login device and directory
- Owner

Although this guide refers to this account as the Oracle Database account or Oracle account, you can assign any name or number to the account after taking into account UIC restrictions. The Oracle Database account owns the run-time libraries and executable images for every Oracle product. Therefore, the database administrator (DBA) must manage this account and install all Oracle products from it.

> **Note:** In *Oracle Database Administrator's Guide*, the Oracle Database account is sometimes referred to as the DBA account. The Oracle Database account is not the same as the SYS or SYSTEM database user names that are created for every database. It is an HP OpenVMS account name.

The UIC group number of the Oracle Database account must be greater than the system parameter MAXSYSGROUP (which defaults to octal 10). Group numbers 1 through the value of MAXSYSGROUP are reserved for use by the HP OpenVMS operating system.

If the UIC group number is not greater than the value of MAXSYSGROUP, then all Oracle Database account commands are not supported. For example:

- STARTUP
- CREATE DATABASE or CREATE TABLESPACE
- ALTER DATABASE or ALTER TABLESPACE

The following command gives the value of MAXSYSGROUP (in decimal format):

```
$ WRITE SYS$OUTPUT F$GETSYI("MAXSYSGROUP")
```

2.4.2 Adding a Record in the User Authorization File

Use the AUTHORIZE utility to create or modify records in the UAF as follows:

1. To run the AUTHORIZE utility, enter the following commands:

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
```

2. At the UAF prompt, enter the ADD command to create the user account from which you install the product software:

```
UAF> ADD /ACCOUNT=ORACLE11G /PASSWORD=ORACLE/UIC=[277,100]-
/DEVICE=device/DIRECTORY=[ORACLE11G]/OWNER="ORACLE DBA"
```

In this example, the account name is ORACLE11G. Ensure that the UIC group number is 277. The UIC group number must be greater than MAXSYSGROUP.

After adding the account, you must alter the account privileges and quotas. This can be done in any order. However, in most cases, the account privileges are set before the account quotas.

> **See Also:** The Hewlett-Packard documentation for more information about using the AUTHORIZE utility

2.4.3 Setting Account Privileges

The following privileges are required as both authorized and default privileges for the Oracle Database DBA account:

> **Note:** Section 2.4.7, "Security Issues with Multiple Databases" explains an exception.

- CMKRNL
- GROUP
- GRPNAM
- IMPERSONATE
- LOG_IO
- NETMBX
- OPER
- PFNMAP
- PRMGBL
- PRMMBX
- SHARE
- SYSGBL
- SYSLCK
- SYSNAM
- SYSPRV

- TMPMBX
- WORLD

Use the AUTHORIZE utility to set the account privileges for the Oracle Database DBA account. At the UAF prompt, use the MODIFY command to add the required default and authorized privileges as follows:

```
UAF> MODIFY ORACLE11G -
/PRIVILEGE=(CMKRNL,NETMBX,OPER,PFNMAP,PRMGBL,PRMMBX,SHARE,SYSGBL, -
  SYSNAM,TMPMBX,GROUP,GRPNAM,IMPERSONATE,LOG_IO,WORLD,SYSLCK, SYSPRV) -
/DEFPRIVILEGE=(CMKRNL,NETMBX,OPER,PFNMAP,PRMGBL,PRMMBX,SHARE, SYSGBL, -
  SYSNAM,TMPMBX,GROUP,GRPNAM,IMPERSONATE,LOG_IO,WORLD,SYSLCK, SYSPRV)
```

2.4.4 Setting Account Quotas

Use the AUTHORIZE utility to change account quotas to meet the requirements of Oracle Database installation.

Account Quotas

Table 2–2 lists the Oracle Database account quotas, their minimum recommended values, and their equivalent HP OpenVMS quota names as displayed by the SHOW PROCESS/QUOTA and SHOW WORKING_SET DCL commands.

Note: These quotas depend on the number of log files, databases, network connections, and other logicals on the system. You may need to customize these values.

Table 2–2 Oracle Database Account Quotas

Account Quota	Minimum Value	Quota Name
ASTLM	250 (the default)	Asynchronous system trap limit
BYTLM	750000	Buffered I/O limit
ENQLM	2000 (the default)	Enqueue quota
FILLM	2048	Open file quota
JTQUOTA	8192	Job table quota
MAXDETACH	0 (the default)	Max detached processes
MAXJOBS	0 (the default)	Max active jobs
PGFLQUOTA	2 million (recommended)	Paging file quota
PRCLM	32	Subprocess quota
WSDEFAULT	2048	Working set default
WSEXTENT	8192	Working set extent
WSQUOTA	4096	Working set quota

Modifying the Default Quotas

After adding a record with the default quotas, use the MODIFY command to alter the default values. The following example changes the enqueue quota (ENQLM) from the default value to a value of 150:

```
UAF> MODIFY ORACLE11G/ENQLM=150
```

2.4.5 Process Rights for Database Administrators

The Oracle Database DBA account must be granted one or more process rights identifiers. These identifiers enable you to run the CONNECT /AS SYSDBA command that is required to perform database administration functions. Table 2–3 displays the combinations of adding and granting rights identifiers to this account:

Table 2–3 Adding and Granting Process Rights Identifiers

Adding...	And Granting...	Enables Control of...
ORA_DBA	ORA_DBA	Any database instance, provided it does not also have an ORA_*sid*_DBA identifier defined for it.
ORA_*sid*_DBA	ORA_*sid*_DBA	Database instance sid only
ORA_*sid*_DBA	ORA_DBA	All database instances except sid

For example, to grant the ORA_*sid*_DBA rights identifier to the Oracle Database account for an instance called TEST, enter the following command:

```
UAF> ADD/IDENTIFIER ORA_TEST_DBA
```

Then, grant the rights identifier to the Oracle Database account as follows:

```
UAF> GRANT/IDENTIFIER ORA_TEST_DBA ORACLE11G
```

If you add the ORA_TEST_DBA rights identifier, but grant only ORA_DBA to the ORACLE11G account, then the account would have insufficient privileges to administer the TEST instance.

The third row of the table shows that you can add and grant different identifiers, which restricts control of a particular instance while it grants control to other instances.

You can also grant database maintenance privileges to accounts (for example, privileges to start and shut down the database) other than the Oracle Database account. However, with the Oracle Database account, the user UIC group number must be greater than MAXSYSGROUP. By granting the ORA_*sid*_DBA identifier, you can similarly restrict user privileges to an instance named *sid*.

2.4.6 Process Rights for SGA Protection

The SGA is protected by an access control list (ACL) that grants access to the identifier ORA_SGA. Before you start a database, add this rights identifier to the UAF, if it does not exist. To do this, the system manager must enter the following command:

```
UAF > ADD/IDENTIFIER/ATTRIBUTES=SUBSYSTEM ORA_SGA
```

Caution: Do not grant the ORA_SGA rights identifier to any user. Oracle software manages the assignment of the rights identifier when it starts the database.

2.4.7 Security Issues with Multiple Databases

If a site has several databases managed by different DBAs, then you may not want to grant the CMKRNL privilege to every DBA Oracle Database account. This privilege enables a DBA to activate any process rights identifiers and to install shared images.

For security reasons, Oracle recommends that the HP OpenVMS system administrator handle these tasks. This person should be responsible for starting Oracle Database instances from a controlled account with a UIC group number that is greater than the value of MAXSYSGROUP. If this is not possible, then at least one Oracle Database account must have the CMKRNL privilege.

2.4.8 Completing Account Setup

Exit the AUTHORIZE utility. If the user account that you modified is logged on at the time you exit this utility, then this user must log out and log in again before the changes take effect.

2.4.9 Verifying Privileges

Enter the following command to verify that your account has the correct privileges and rights identifiers:

```
$ SHOW PROCESS/PRIVILEGE
```

2.4.10 Setting Permissions for File Creation

Set default file protections for the ORACLE11G account to ensure that GROUP and WORLD have READ and EXECUTE permissions, but not WRITE permission on installed files.

1. Enter the $ SHOW PROTECTION command to check the current setting.
2. If the $ SHOW PROTECTION command does not show the expected protection, then set protection in the LOGIN.COM of the ORACLE11G account as follows:

   ```
   $ SET PROTECTION=(s:RWE, o:RWED, g:RE, w:RE)/default
   ```

2.5 SYSGEN Parameters

> **See Also:** READMEVMS.DOC in the RDBMS directory for instructions on calculating MIN_GBLPAGES, and MIN_GBLSECTIONS

You can create and use reserved memory on an HP OpenVMS system. The following sections describe the utilities that you can use to set the values of the various SYSGEN parameters:

- Section 2.5.1, "Reserved Memory Registry"
- Section 2.5.2, "AUTOGEN"

2.5.1 Reserved Memory Registry

Through its interface within the SYSMAN utility, the reserved memory registry enables you to configure an HP OpenVMS system with large amounts of memory set aside for use within memory-resident global sections. The AUTOGEN utility considers the preallocated reserved memory when it tunes the system.

Oracle recommends that you do not use reserved memory until the instance SGA size is stable. In most cases, this is a few weeks after the database is performing well.

The advantages to reserving memory for an SGA are as follows:

- You can be certain that the memory is available and that the system is correctly tuned.

- The memory is reserved at startup time as contiguous aligned physical pages. This enables the system to optimize the mapping of the SGA.
- The memory is preallocated and zeroed. This results in faster SGA mapping and faster instance startup.

To reserve memory for an SGA, use the SYSMAN utility. The size qualifier is specified in megabytes. For example, to reserve memory for a 6 GB SGA for the SID named TEST, use the following commands:

```
$ MCR SYSMAN
SYSMAN> RESERVED_MEMORY ADD ORA_TEST_SGA/SIZE=6144/ALLOCATE/ZERO/PAGE
SYSMAN> EXIT
```

Next, run AUTOGEN and restart the system to allow AUTOGEN to adjust other system parameters for the reduced amount of memory that is available to the rest of the system.

Memory for multiple SGAs may be reserved. Any change to the name or size of a piece of reserved memory may require restarting the system. During instance startup, Oracle Database compares the size of the reserved memory, if any, with the size of the SGA. If appropriate, one of the following messages is included in the alert log:

```
** Reserved memory size = size greater than created SGA size = size**
** Please reduce reserved memory size to avoid wasting memory. **

** Memory was not reserved for the SGA. SGA size = size **
** There might be performance advantages to allocating memory for
    the SGA in the VMS reserved memory registry. **
```

If the memory reserved for an SGA is insufficient, then the instance startup fails. In this case, you must adjust the amount of reserved memory, run AUTOGEN, and restart the system. For example, to expand the SGA to 6.5 GB for the TEST instance, enter the following commands:

```
$ MCR SYSMAN
SYSMAN> RESERVED_MEMORY REMOVE ORA_TEST_SGA
SYSMAN> RESERVED_MEMORY ADD ORA_TEST_SGA/SIZE=6656/ALLOCATE/ZERO/PAGE
SYSMAN> EXIT
```

Next, run AUTOGEN and save it permanently using the SAVEPARAMS parameter. This saves the change in the AGEN$FEEDBACK.DAT file.

If you do not want to save permanently then, instead of running AUTOGEN, use SYSMAN to free the memory reservation and proceed with starting the Oracle instance without reserved memory. The risk involved with this is that the system may not have enough fluid pages to create the SGA. In addition, the performance advantages of using reserved memory are not available.

> **See Also:** The Hewlett-Packard documentation for more information about the reserved memory registry

To review memory reservations on the node, run the following command:

```
$ SHOW MEMORY/RESERVED

System Memory Resources on 13-JUL-2001 09:57:11.72
Memory Reservations (Pages)    Group    Reserved   In Use   Type
ORA_PROD_SGA                   SYSGBL   5120       0        Allocated
ORA_PROD_SGA                   SYSGBL   5          0        Page Table
Total (40.04 Mbytes reserved)           5120       0
```

2.5.2 AUTOGEN

Because SYSGEN parameters affect the entire operating system, the system administrator is the only person who should modify them. At some sites, the DBA and system administrator may be the same person. The instructions in this section are directed to the system administrator.

For setting or modifying SYSGEN parameters, HP OpenVMS provides the AUTOGEN utility. You can also use SYSGEN, but this is an older utility and its use is discouraged, except perhaps for checking current values. You must have the SYSPRV or BYPASS privilege to run these utilities.

AUTOGEN provides a permanent method for setting parameters, and it documents all changes. AUTOGEN also lets you recalculate any parameters that depend on other parameters that you may have changed. Remember to record parameter values before changing them, and determine in advance what results you expect from the changes you make. If the expected changes do not occur, then restore the old values before trying again.

The procedure is as follows:

1. Run the AUTOGEN utility with the SAVPARAMS parameter by entering the following command:

   ```
   $ @SYS$UPDATE:AUTOGEN SAVPARAMS GETDATA
   ```

   This step saves current parameters to a file named AGEN$FEEDBACK.DAT.

2. If you have not already done so, run AUTOGEN GETDATA to ensure that the file PARAMS.DAT has been generated.

   ```
   $ @SYS$UPDATE:AUTOGEN GETDATA
   ```

3. Examine the parameter settings in the current PARAMS.DAT, and if they are correct, then change them in the MODPARAMS.DAT file by using the MIN_*parameter=value* format as follows:

   ```
   MIN_GBLSECTIONS=3000
   MIN_GBLPAGES=300000
   ```

   The MIN prefix indicates that you are setting a lower limit for the specified parameter. To use the MIN prefix, identify the current value of the specified parameter (using SYSGEN) and the amount by which you want to increase the parameter. The sum of these two values is used for the MIN_*parameter* entry.

4. Run the AUTOGEN utility with the GENPARAMS and REBOOT parameters by entering the following command:

   ```
   $ @SYS$UPDATE:AUTOGEN GENPARAMS REBOOT
   ```

   Caution: The REBOOT parameter in this example causes the system to automatically restart when the SYSGEN is complete.

   This step generates new SYSGEN parameters in a file named SETPARAMS.DAT and runs the SYSGEN utility to set these parameter values as specified in the file.

   See Also: *HP OpenVMS System Management Utilities Reference* for more information about using the AUTOGEN utility

2.6 System Configuration

If the system fails to satisfy any listed requirement, then perform the tasks listed in the next section as necessary to configure the system to meet these requirements.

Table 2–4 lists the requirements for installing Oracle Database.

Table 2–4 HP OpenVMS System Configuration Summary

System Factors	Requirements for Oracle Database
HP OpenVMS Parameters: Minimum Global Pages or Sections	Verify that the values for MIN_GBLPAGES and MIN_GBLSECTIONS are greater than, or equal to, the values that were calculated for these parameters, as documented in READMEVMS.DOC.
HP OpenVMS Accounts	You need an HP OpenVMS account that is dedicated solely to installing and upgrading Oracle products. The instructions in this book are provided with the assumption that the account is called ORACLE11G.
Permissions for File Creation	Default Protections

2.7 Performing Setup Tasks As the SYSTEM User

Log in as the SYSTEM user and perform the following tasks to set up the environment for Oracle Database:

- Section 2.7.1, "Creating an Oracle Database Account"
- Section 2.7.2, "Creating an Apache Server Account"

2.7.1 Creating an Oracle Database Account

The Oracle Database account is the HP OpenVMS user account that owns Oracle Database software after installation. Run Oracle Universal Installer with this user account.

Details of creating this account are covered in Section 2.4, "Creating a User Account"

Use the AUTHORIZE utility to create an oracle account with the properties listed in Table 2–5.

Table 2–5 Properties of the Oracle Database Account

Account	Properties
Login Name	Any name, but this document refers to it as the Oracle Database installation owner. For example, ORACLE11G.
UIC	The group UIC number, which must be greater than MAXSYSGROUP.
Home Directory	Select a login directory consistent with other users' login directories. The login directory of the ORACLE11G account should not be used as the ORA_ROOT/ORACLE_HOME directory.

Caution: Use the Oracle Database account only for installing, maintaining and managing Oracle software. Do not use it for purposes unrelated to Oracle Database server. Do not use SYSTEM as the ORACLE11G account.

Sites with multiple ORA_ROOT directories on one system may install Oracle software with the same ORACLE11G account, or separate accounts. Each ORACLE11G account must have the same group UIC.

2.7.2 Creating an Apache Server Account

The Apache account is an HP OpenVMS user account that owns the Apache server after installation. If you use a default Apache configuration (one that listens to ports lower than 1024, which are reserved for the system), then Oracle recommends, for security reasons, that a separate account owner be set up for Apache. Configure the Apache server to assign the ownership of listener and module actions to this account. It is not necessary for the Apache user account to have privileges to run the Apache Server on port numbers that are lower than 1023. Images that are installed with privileges during installation in the oracle account ensure that this can be done.

See Section G.3.3, "Customizing the Server Environment" for information about how to configure the server to run as a user other than the 'oracle' user.

The Apache account owner should have minimal user privileges, and should not be a member of any groups where files are not intended to be visible to the public.

2.8 Performing Setup Tasks As the oracle user

Log in to the oracle Database account and perform the following tasks as necessary:

- Section 2.8.1, "Setting the Display"
- Section 2.8.2, "Setting Up a Batch Queue for the Oracle Enterprise Manager Agent and HAS"

2.8.1 Setting the Display

Before starting Oracle Universal Installer, set the display by using the following command:

```
$ SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
```

Verify that you have set the display correctly by using the following command:

```
$ RUN SYS$SYSTEM:DECW$CLOCK
```

Do not use the host name or IP address of the system where you install the software unless you perform the installation from the X Window console of that system. Use the computer name or IP address of the workstation if you are installing from a remote system. If you are not sure of the X server and screen settings, then use 0 (zero) for both.

2.8.2 Setting Up a Batch Queue for the Oracle Enterprise Manager Agent and HAS

In this release, the architecture and implementation of the Oracle Enterprise Manager Agent has undergone a major revision. The new agent is based on an HTTP listener that communicates with the Management Server using XML and that uses Perl to perform a number of tasks. This has been implemented to run in a batch queue instead of running as detached processes. The Oracle Enterprise Manager Agent runs a main process. To support monitoring of both internal (using the same Oracle home) and external databases and listeners, the Enterprise Manager Agent submits additional batch jobs.

From the operational and management perspective, this has been implemented to run under a batch queue to get a better collective view of all processes related to the Enterprise Manager Agent running at any given time. This makes it easy to monitor and control all processes spawned by the Enterprise Manager Agent. In addition, it may be possible to delete batch jobs or processes that can potentially fail when trying to connect to databases or listeners that are not running.

> **Note:** Although the Enterprise Manager Agent is no longer installed as part of a database installation, it is available as part of the separate Enterprise Manager Grid Control release.

For High-Availability Services (HAS), OCSSD, EVMD, and CRSD, are collectively run as batch process. The Enterprise Manager Agent and HAS use the following method to determine the name of the batch queue on which to submit jobs:

- If the `ORA_BATCH_QUEUE_`*`hostname`* logical name is defined, then the value of this logical is used.
- If the `ORA_BATCH_QUEUE_`*`hostname`* logical is itself a valid queue name, then it is used.

The `ORA_BATCH_QUEUE_`*`hostname`* logical name need not always be defined. It is sufficient that it is defined at least once during installation. Alternatively, you can define it before running `ORAUSER.COM` at any time after the installation. Ensure that the Enterprise Manager Agent and HAS services can only be started up after setting up this batch queue logical name and running `ORAUSER.COM`.

The two components record the batch queue name in the installation, and use that as a reference for subsequent calls to the services. The default batch queue `SYS$BATCH` generally points to a batch queue that runs under a lower priority (usually, 2) than the priority of a normal user (usually, 4). Therefore, it is inappropriate to use the default queue for the two services. This would result in poor and, in some cases, defective performance of the two services.

> **Note:** Oracle recommends to use a dedicated batch queue.

The number of jobs submitted for the Enterprise Manager Agent batch queue depends on the number of monitoring targets which include databases and listeners. An approximation of two to five monitoring targets would be a good number for the `job_limit`. In the case of HAS, a `job_limit` of about 10 would be sufficient. The batch queue priority should be the same as all other Oracle processes, which is typically 4 on a standard VMS system.

2.9 Setup Tasks for Oracle Products

The following products require preinstallation steps to be completed before you install Oracle Database software:

- Section 2.9.1, "Precompilers and Tools"
- Section 2.9.2, "Network and System Management Products"

2.9.1 Precompilers and Tools

The following sections list the preinstallation tasks for the precompilers and tools that are required for HP OpenVMS.

2.9.2 Network and System Management Products

This section describes tasks that need to be completed prior to installation if you have existing network and system management products.

2.9.2.1 Setup Tasks for Oracle Net Services

If you have an existing installation of Oracle Net Services running on the same TCP ports, then shut down all listeners before installing it. To determine if the TCP port is in use, enter the following command:

```
$ TCPIP SHOW DEVICE/PORT=port_number
```

If the port is in use, a device name will be displayed. Issue the following command:

```
$ SHOW DEVICE/FULL device_name
```

The output of this command will contain a process ID. Issue the following command:

```
$ SHOW PROCESS/CONTINUOUS/ID=pid
```

The full path to the program that is running and which is using the port will be indicated in this display. This path includes the Oracle home from which you should stop the Listener.

2.10 Identifying Required Software Directories

You must identify three directories for the Oracle software as described in the following sections:

- Section 2.10.1, "Oracle Inventory Directory"
- Section 2.10.2, "Oracle Base Directory"
- Section 2.10.3, "Oracle Home Directory"

The following subsections describe the requirements for these directories.

2.10.1 Oracle Inventory Directory

The Oracle inventory directory (`ORAINVENTORY`) stores an inventory of all Oracle software installed on the system. It is required by, and shared by, all Oracle software installations on a single system.

The Installer creates this directory if it does not exist and sets the correct owner, group, and permissions on it.

Note: Oracle recommends that you specify the inventory location by creating the `oraInst.loc` file in the `sys$login`:[*node_name*] directory. Create this directory if it does not exist. The contents of the `oraInst.loc` file is as follows:

```
inst_group=VMS1
inventory_loc=/DISK1/ORACLE11GR2/INVENTORY_NODE1
```

`VMS1` is the UIC group of the account used to install Oracle. All Oracle software installations depend 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.

2.10.2 Oracle Base Directory

The Oracle base directory is a top-level directory for Oracle software installations. On HP OpenVMS, a typical Oracle base directory is similar to the following example:

```
DISK1:[ORACLE11GR2]
```

2.10.3 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.

Oracle Universal Installer creates the directory path that you specify. It also sets the correct ownership and permissions. You do not need to create this directory.

Oracle Universal Installer can use either one of the following format examples:

- OpenVMS format:

  ```
  DISK1:[ORACLE11GR2.DBHOME1]
  ```

- UNIX format:

  ```
  /disk1/oracle11gr2/dbhome1
  ```

2.11 Guidelines for Placing Oracle Recovery Files

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

Use the following guidelines to place Oracle Database or recovery files:

- To prevent disk failure from making both the database files and the recovery files unavailable, place the recovery files on a different physical disk from the database files.
- The disk that you select must have at least 2 GB of free space.

  The disk space requirement is the default disk quota configured for the flash recovery area. The default disk quota is specified by the `DB_RECOVERY_FILE_DEST_SIZE` initialization parameter. If you select the Custom installation type, then you

can specify a different disk quota value. After you create the database, you can also use Oracle Enterprise Manager Grid Control to specify a different value.

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

- The Oracle Database account must have write protection to create the files in the specified path.

2.12 Creating Directories for Oracle Database or Recovery Files

Use the following guidelines when deciding the location of Oracle Database or recovery files:

- You can choose either a single device or more than one device to store the database files depending on the following criteria:
  - If you want to use a single device, then choose a device on a physical device that is dedicated to the database.

    For performance and reliability, choose a redundant array of independent disks (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 device, then choose devices that are dedicated to the database.

    This method enables you to distribute physical I/O and create separate control files on different devices for increased reliability. 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, the device (or devices) that you choose must have at least 1.2 GB of free disk space.

  For production databases, you must estimate the disk space requirement depending on the use you want to make of the database.
- For optimum performance, the devices that you choose should be used only by the database.
- The Oracle Database account must have write permissions to create the files in the path that you specify.

Creating Required Directories

> **Note:** You must complete this procedure only if you want to place Oracle Database or recovery files on a separate device instead of placing them in the Oracle base directory.

To create directories for Oracle Database or the recovery files on separate devices:

1. Use the `SHOW DEVICE` *`device_name`* command to determine the free disk space on each mounted device.
2. From the display, identify the devices to use. Table 2–6 lists the device requirements for recovery files.

Table 2–6 Device Requirements for Recovery files

File Type	Device Requirements
Data files	Choose either: ■ A single device with at least 1.2 GB of free disk space ■ Two or more devices with at least 1.2 GB of free disk space in total
Recovery files	Choose a device with at least 2 GB of free disk space.

If you are using the same device 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 root device directories for the devices that you identified.

2.13 Stopping Existing Oracle Processes

Note: If you are installing additional Oracle Database 11*g* Release 2 (11.2) 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 Services listener using TCP/IP port 1521 and the IPC key value `EXTPROC`. However, if an existing Oracle Net Services listener process uses the same port or key value, Oracle Universal Installer can only configure the new listener, but 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 if an existing listener process is running, and to shut it down if necessary:

1. Log in as the Oracle Database account user.
2. Enter the following command:

```
$ PIPE SHOW SYSTEM/OWN=ORACLE | SEARCH SYS$INPUT last_8_chars_of_listener_name
```

This command displays the Oracle processes that are running in the `oracle` account. Run the following command to identify the Oracle processes that run as listener processes:

```
$ WRITE SYS$OUTPUT F$GETJPI(pid,"IMAGNAME")
```

For each process that is running as a listener process, run the following command to determine the Oracle home in which it is running:

```
$ SHOW PROCESS/CONTINUOUS process_name
```

In this example, *process_name* is the name of the listener process that is indicated by the command that you just entered.

Note: If no Oracle Net Services listeners are running, then refer to Section 2.14, "Configuring Oracle User's Environment" to continue.

3. Set up the Oracle environment for the Oracle home area where the listener is running as follows:

```
$ SET DEFAULT Oracle_home_dir
$ @ORAUSER
```

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

```
LSNRCTL STATUS listenername
```

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

5. If the TCP/IP port number determined in the preceding step is 1521, or the IPC key value is EXTPROC, then enter a command similar to the following:

```
LSNRCTL STOP listenername
```

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

See Also:

- Appendix H for a list of commands and that you can use when performing this procedure
- *Oracle Database Administrator's Reference for HP OpenVMS Itanium* for information about stopping existing Oracle processes on Oracle RAC

2.14 Configuring Oracle User's Environment

Before you start Oracle Universal Installer, you must configure the Oracle Database installation owner environment.

To achieve this, perform the following steps:

1. Start a new terminal session.
2. Set up the display using the SET DISPLAY command.
3. Set up the inventory directory, and create the ORAINST.LOC file. This step is optional but recommended. If you do not create the inventory directory, then an ORAINST.LOC file is created under the sys$login:*node_name* directory. The default inventory location is the directory INVENTORY_*node_name* created as the top-level directory of ORACLE_HOME.
4. Define the logical name ORA_BATCH_QUEUE_*nodename* to point to an existing batch queue. Oracle recommends that you use a dedicated Batch Queue.

> **See Also:** Appendix C, "Installing, Configuring, and Running Enterprise Manager Agent" for more information about batch queues

5. Create a file, CLUSTER_NODES.DAT (if you are installing Oracle Clusterware), in the SYS$LOGIN directory with the list of nodes that are to comprise the Oracle RAC cluster. This file displays each node on a separate line with the primary node listed first.
6. Enter the $ SET NOVERIFY command.

7. Enter the `$ SET PROCESS/PRIV=ALL` command.
8. To start the `RUNINSTALLER` refer to Section 3.3, "Installing Oracle Database Software"

2.15 The ORATAB File

The `ORATAB` file contains information about all Oracle installations and databases installed on the node. This information is used to discover Oracle product installations. When an Oracle product is installed, this file is updated with information about the new Oracle home. If this file is not found, then a new `ORATAB` file is created.

To check if the `ORATAB` file exists:

1. Check for the `ORATAB` file in the *`hostname`* subdirectory of `SYS$LOGIN`. For example:

   `DISK:[mylogin.hostname]ORATAB.`

2. If the `ORATAB` file is not found at the location described in Step 1, then use the file in the `SYS$LOGIN` directory. For example:

   `DISK:[mylogin]ORATAB.`

3. If the `ORATAB` file is not found in any of the locations mentioned in the previous steps, then it means there is no `ORATAB` file.

3

Installation Tasks

This chapter describes how to start Oracle Universal Installer and install Oracle products on the system. It includes information about the following topics:

- Section 3.1, "Installation Overview"
- Section 3.2, "Downloading Oracle Software"
- Section 3.3, "Installing Oracle Database Software"
- Section 3.4, "Reviewing the Log of an Installation Session"
- Section 3.5, "Cleaning Up After a Failed Installation"

> **Note:** Review the information in Chapter 1, "Installation Overview" and complete the tasks listed in Chapter 2, "Preinstallation Tasks" before beginning the installation.

3.1 Installation Overview

Oracle Database software is available on Installation media or you can download it from the Oracle Technology Network website. 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 noninteractive installations without using the GUI. See Appendix B for information about noninteractive installations.

This chapter describes how to install Oracle Database from the product media or from the hard disk. 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 discs if you have them. Perform one of the following tasks, depending on the method that you want to use:

- See Section 3.3.1, "Installing Oracle Database from the Hard Drive" to copy the software to a hard disk.
- See Section 3.2, "Downloading Oracle Software" to download the software from Oracle Technology Network.
- See Section 3.3, "Installing Oracle Database Software" to install the software from disc or from an existing hard disk location.

3.1.1 Reviewing Product-Specific Installation Guidelines

Review the following guidelines before starting Oracle Universal Installer:

- Oracle Universal Installer

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

- Oracle Label Security

  To install the Oracle Label Security option, you must select a Custom installation type.

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

  If you want to configure Oracle Label Security to use Oracle Internet Directory, then you must select the Oracle Internet Directory option when running Database Configuration Assistant.

- Products not installed by default: choose **Custom** installation type. These products are:

  Oracle Label Security

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

3.2 Downloading Oracle Software

To download the Oracle Database 11*g* Release 2 (11.2) Media Pack, perform the steps provided in the following sections:

- Section 3.2.1, "Ordering Oracle Database Media"
- Section 3.2.2, "Downloading Management Agent Software"

See Also: *OpenVMS: FAQ for Oracle RDBMS Release 11.2 on HP OpenVMS* (document 2012766.1)

3.2.1 Ordering Oracle Database Media

The Oracle OpenVMS kit 11*g* Release 2 (11.2) is available for download on Oracle Technology Network and eDelivery. To order DVD media, see document 1071023.1 on the My Oracle Support website at

`https://support.oracle.com`

To locate document 1071023.1, perform the following steps:

1. Log on to My Oracle Support.
2. Set Article ID as the search source.
3. Enter 1071023.1 in the search field, then click **Search**.

The Oracle Database Media Pack for Oracle Database 11*g* Release 2 (11.2) for HP OpenVMS on request, contains the product for Oracle 11*g* Release 2 (11.2.0.4.0) for HP OpenVMS Itanium.

3.2.2 Downloading Management Agent Software

For HP OpenVMS 11*g* Release 2 (11.2), the 10.2.0.2 Enterprise Manager Agent should be utilized. To download Management Agent, select the required HP OpenVMS platform from the following website at

http://www.oracle.com/technetwork/oem/grid-control/downloads/agentsoft-090381.html

3.3 Installing Oracle Database Software

You can install Oracle Database either from the hard disk or from installation media. The following sections describes the methods for installing Oracle Database from these media:

- Section 3.3.1, "Installing Oracle Database from the Hard Drive"
- Section 3.3.2, "Installing Oracle Database from Install Media"

3.3.1 Installing Oracle Database from the Hard Drive

You can avoid the need to mount and unmount installation media during installation by copying the contents of each installation medium to an On-Disk Structure Level 5 (ODS-5) formatted disk. You must have enough disk space available, depending on the type of install.

1. Set up an X Window System environment by using the following DCL command:

   ```
   $ SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
   ```

2. Copy the install media files onto a hard drive. Create a directory to hold the contents of the DVD on an `ODS-5` disk. For example, you may perform the following steps, replacing `DISK$VMS` with an appropriate `ODS-5` disk name, `DQA0` with the name of the DVD drive, and `ORACLE` with the name of the Oracle Database account:

   ```
   $ CREATE/DIR DISK$VMS:[11GKIT]/OWNER=ORACLE11G
   $ SET DEF DISK$VMS:[11GKIT]
   $ MOUNT/OVER=IDENT DQA0 ! Mount DVD
   $ SET PROC/PARSE=EXTENDED
   $ COPY/READ/WRITE DQA0:[000000...]*.*;*[...]/EXCLUDE=*.SYS
   $ DISMOUNT DQA0
   ```

   The software is now staged on disk and ready for installation using Oracle Universal Installer.

3. Start Oracle Universal Installer from the `[11GKIT.DATABASE]` directory on the hard drive. For example:

   ```
   HOST25> SET DISPLAY/CREATE/TRANSPORT=TCPIP/node=host01/server=3
   HOST25> @DISK$VMS:[11GKIT.DATABASE.DISK1]runInstaller.com
    - checking OpenVMS ECO status

   Do you wish to continue? (Y/N) [Y] : y

    - checking OpenVMS quota settings
   OpenVMS quota setting checked successfully.

   Please specify the location for ORACLE_BASE
   This must be a directory on an ODS-5 formatted disk
    and must NOT be within/under ORACLE_HOME
   ```

```
Enter ORACLE_BASE: disk66:[Oracle11204]

Please specify the location to be used for ORACLE_HOME

ORACLE_HOME location: disk66:[Oracle11204.EEsvr]

WARNING: The specified ORACLE_HOME does not exist

         disk66:[Oracle11204.EEsvr]

Create now? [N] : Y

Creating directory ...
%CREATE-I-CREATED, disk66:[Oracle11204.EEsvr] created


Please specify the location for temporary files created during installation.
This must be a directory on an ODS-5 formatted disk.

Enter scratchPath: disk67:[scratch2]


WARNING: The specified scratchPath does not exist:

         disk67:[scratch2]

Create now? [N] : Y

Creating directory ...
%CREATE-I-CREATED, disk67:[scratch2] created

%DCL-I-SUPERSEDE, previous value of ORA_AUTO_INSORACLE has been superseded

Starting Oracle Universal Installer...

Preparing to launch Oracle Universal Installer from
/disk67/scratch2/OraInstall<Date and Time>. Please wait ...

Executing: @disk67:[scratch2.OraInstall<Date and Time>]setup.com
Oracle Universal Installer, Version 11.2.0.4.0 Production
Copyright (C) 1999, 2013, Oracle. All rights reserved.

Unknown host: _WSA14
```

4. Complete Steps 3 through 7 in Section 3.3.2.2, "Installing Oracle Database" to continue with the installation process.

3.3.2 Installing Oracle Database from Install Media

Set up an X Window System environment by using the following DCL command:

```
$ SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
```

The following procedure describes the installation of products with the use of the Install Media files and Oracle Universal Installer:

- Section 3.3.2.1, "Mount the DVD"
- Section 3.3.2.2, "Installing Oracle Database"

- Section 3.3.2.3, "Completing Oracle Database Server Enterprise Edition Installation"

3.3.2.1 Mount the DVD

Mount the Oracle Installation DVD. For example:

```
$ MOUNT/OVERRIDE=IDENTIFICATION ddcn
```

where:

ddcn is a valid DVD drive.

Caution: Oracle does not support Oracle Universal Installer that is shipped with any prior release to install Oracle Database 11*g* Release 2 (11.2) components.

3.3.2.2 Installing Oracle Database

To install Oracle Database from the DVD:

1. Log in to the Oracle Database account.
2. Start Oracle Universal Installer from the `DATABASE` directory on the DVD with the following command:

   ```
   $ @ddcn:[DATABASE]RUNINSTALLER
   ```

   After Oracle Universal Installer is started, the Welcome screen is displayed. Click **Next** to continue the installation.
3. An introductory screen appears which allows you to uninstall previously installed Oracle Products.
   a. Click **Next** to continue.
4. The Available Products screen is displayed.
   a. Select Oracle Database Server or Oracle Client.
   b. Click **Next** to continue.
5. The Installation Types screen is displayed.
   a. Select an installation type: Enterprise Edition, Standard Enterprise, or Custom.

      Oracle Database Custom Installation: Enables you to select the individual components to install from a list of available components.

      If you select the Custom Installation option, the Available Product Components screen is displayed. It lists all the products and components that are available for installation. A typical custom installation configuration is selected by default. Some actions that may differ from a standard installation are:

Select the products that you want to install, or deselect the products that you do not want to install, and click **Next**.
Provide responses to any screen prompts that are displayed. With custom selection of products, you may want to change default values for items such as: port numbers or product-specific directory names. In the end, the Summary screen is displayed.

   b. Click **Next.**

      The Database Server Summary page is displayed.

c. Click **Install**.

6. The Install screen is displayed. This screen indicates installation progress. Any errors or warnings are displayed in a dialog box where you can select to fix and retry, to ignore and continue, or to quit.

   a. If you chose to install **Software Only,** then the End of Installation screen is displayed next.

   b. During installation, the following messages regarding linking appears in the terminal screen:

```
Using hex value: 5A6E7EB2 for install.id
  - unzipping LinkScripts_client.zip
  - unzipping LinkScripts_server.zip

Starting LINK operation.   This could take a while. The linklog information
will be written to ora_root:[install]make.log. Please wait ...

Completed LINK operation.

Deleting ORACLE runtime environment...
Dropping "SERVER_5A6E7EB2" from LNM$FILE_DEV list.
Redefining LNM$FILE_DEV...
ORACLE runtime environment deleted.
EXECUTING RELINK DEPLOY
   - deploying recently linked executables to ORA_ROOT:[BIN]
   - files deployed:   196 / 196

Completed RELINK DEPLOY
```

   c. When configuration is complete, the End of Installation screen is displayed.

7. From the End of Installation screen, you may exit or install additional products.

   a. This screen is displayed when the installation is complete. You may choose to exit Oracle Universal Installer or to begin another installation:

   b. Click **Exit** to end Oracle Universal Installer session.

   c. Click **Next Install** to begin another installation without stopping and restarting Oracle Universal Installer.

3.3.2.3 Completing Oracle Database Server Enterprise Edition Installation

If no errors occur during installation, then the installation is complete. To create a database instance, use Database Configuration Assistant.

See Also: Section 4.1, "Database Configuration Assistant"

3.4 Reviewing the Log of an Installation Session

Oracle Universal Installer creates the `ORAINVENTORY` directory the first time that it is run on a computer. It does this to keep an inventory of products that it installs on the system and an inventory of other installation information.

The location of the `ORAINVENTORY` directory is defined in the `ORAINST.LOC` file, which is located in the nodename sub directory which refers to the node where the install was done. An inventory is created the first time that an Oracle Universal Installer installation is performed on a particular system. The inventory is then detected and referenced during subsequent installation sessions.

Note: If the same node is used, the ORAINST.LOC is detected and referenced only during subsequent installation sessions. If on a cluster, this represents a problem.

Oracle suggests that you create additional nodename directories and copy the ORAINST.LOC file to them so that any node can subsequently be used for additional install actions.

The log file of the most recent installation is INSTALLACTIONS.LOG located in the [.LOGS] subdirectory of the ORAINVENTORY directory.

INSTALLACTIONS*date_time*.LOG

For example:

INSTAllACTIONS 2001-02-14_09-00-56-AM.LOG

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

The make.log file in the ORA_ROOT:[INSTALL] directory contains a log of every make file action that is run during the installation process. The make.log file should be examined for any linker errors that may have occurred during the installation and ensure that corrective action is taken. Do not delete or alter the make.log file.

3.5 Cleaning Up After a Failed Installation

If an installation fails, then (before you attempt another installation) you may need to remove files that Oracle Universal Installer created during the last session.

To clean up after a failed installation:

1. Start Oracle Universal Installer.
2. Click **De-install Products** and select any products that were left after the failed installation.
3. Click **Remove**.

To complete the clean up, you may need to manually remove the ORA_ROOT directory.

4

Postinstallation Tasks

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

- Section 4.1, "Database Configuration Assistant"
- Section 4.2, "Required Postinstallation Tasks"
- Section 4.3, "Configuring New or Upgraded Databases"
- Section 4.4, "Changing User Passwords"
- Section 4.5, "Performing Configuration Tasks as the SYSTEM User"
- Section 4.6, "Performing Configuration Tasks as Oracle User"
- Section 4.7, "Required Product-Specific Postinstallation Tasks"

If you install and intend to use any of the products listed in Section 4.7, "Required Product-Specific Postinstallation Tasks" then you must perform the tasks listed in the product-specific subsections.

> **Note:** This chapter describes basic configuration only. See *Oracle Database Administrator's Reference for HP OpenVMS Itanium* and the product-specific administration and tuning guides for more sophisticated configuration and tuning information.

4.1 Database Configuration Assistant

The following sections describe the procedure for starting Database Configuration Assistant:

- Section 4.1.1, "Running Database Configuration Assistant in Interactive Mode"
- Section 4.1.2, "Using Database Configuration Assistant Response Files"
- Section 4.1.3, "Database Configuration Assistant Unsupported Options"

> **Note:** You must use UNIX-based syntax when you specify file and path names for files such as data files, and log files. For example, the HP OpenVMS file specification `DISK$ORA:[ORAR2.DATAFILES]` must be expressed as the UNIX path `/disk$ora/orar2/datafiles`.

4.1.1 Running Database Configuration Assistant in Interactive Mode

To start Database Configuration Assistant in interactive mode, perform the following:

1. Set up the X Window System environment by using the following DCL command:

```
SET DISPLAY/CREATE/TRANSPORT=TCPIP/NODE=workstation_name or IP_address
```

2. Set default to the [BIN] directory in the Oracle home directory, and then enter the following command to start DBCA:

```
$ @disk:[oracle_home_directory]ORAUSER
$ SET DEFAULT ORA_ROOT:[BIN]
$ DBCA
```

Note: Click **Finish** to go to the last screen. This accepts the default answers to the questions on the other screens. However, if DBCA determines that the default answers to one or more screens do not apply, the DBCA will not display the last screen.

Database Configuration Assistant graphical user interface (GUI) is displayed.

See Also: See Appendix B.5, "Running Database Configuration Assistant in Noninteractive or Silent Mode" for information about running Database Configuration Assistant in noninteractive or silent mode.

4.1.2 Using Database Configuration Assistant Response Files

An alternative to creating an argument file to pass command-line options to Database Configuration Assistant is to use a response file. A sample response file (DBCA.RSP) is provided in the ORA_ROOT:[ASSISTANTS.DBCA.LOGS] directory. Make a copy of this file and customize it as necessary. Follow the instructions that are contained in comments. The value that is specified for the CREATE_TYPE field determines which of the other sections are applicable. Each field has a comment header block with the following entries: Name, Datatype, Description, Valid values, Default value, and Mandatory (Yes/No) descriptions.

For example:

```
#_____________________________________________________________________________
# Name          : CREATE_TYPE
# Datatype      : String
# Description   : Type of database creation
# Valid values  : "createDatabase" \ "createTemplateFromDB" \
#                 "createCloneTemplate"
# Default value : None
# Mandatory     : Yes
#_____________________________________________________________________________
CREATE_TYPE = "createDatabase"
```

All mandatory values must be provided, but only for the applicable sections as indicated by the CREATE_TYPE value. After customizing the response file, start Database Configuration Assistant and pass the name of the response file at the command line as follows:

```
$ DBCA -SILENT -RESPONSEFILE full_response_file_name
```

> **Note:** Either HP OpenVMS-based arguments or UNIX-based arguments may be used in fields that are contained within a response file, but the response file argument that is passed to the `DBCA.COM` script at the command line with the `RESPONSEFILE` keyword must be in HP OpenVMS-based syntax.

4.1.3 Database Configuration Assistant Unsupported Options

The following Database Configuration Assistant options are not supported with this release:

- OLAP
- UltraSearch
- Sales History Demonstration

4.2 Required Postinstallation Tasks

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

- Section 4.2.1, "Downloading and Installing Patches"
- Section 4.2.2, "Configuring Oracle Products"
- Section 4.2.3, "Setting the NLS_LANG Environment Variable"

4.2.1 Downloading and Installing Patches

Check the My Oracle Support website for required patches for the installation. To download required patches:

1. Use a Web browser to view the My Oracle Support at

   `https://support.oracle.com`

2. Log on to My Oracle Support.

   > **Note:** If you are not a My Oracle Support registered user, then click **Register** on the home page and follow the registration instructions.

3. On the main My Oracle Support page, click **Patches & Updates**.
4. Select **Product or Family (Advanced)** search.
5. Specify the following information, then click **Search**:
   - In the **Product is** field, choose `RDBMS Server` from the list.
   - In the **Release** field, choose the current release number from the list.
   - Select the **Type** field from the list.
   - Choose **Patchset** from the list.

     Selecting the Patchset option opens the Search results page. This page provides a list of patches that can be downloaded and saved. Clicking a patch enables you to choose options from the options menu on the top of the screen.

6. Open the patch Read Me file, which you can access by clicking the **Read Me** icon in the options menu for each patchset, and follow the installation instructions.

   Some patches install with OUI; others require special procedures. Oracle recommends that you always read the Read Me before proceeding.

7. Return to the Patch Set page, click **Download**, and save the file on the system.

8. Use an `unzip` utility to uncompress the Oracle patches that you downloaded from My Oracle Support.

4.2.2 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, see the appropriate manual in the product documentation library.

4.2.3 Setting the NLS_LANG Environment Variable

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

> **See Also:** Appendix E, "Configuring Oracle Database Globalization Support" for more information about the `NLS_LANG` logical name

4.3 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 may 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 time.

4.3.1 Upgrading 10.2 Databases

Each existing 10.2 Database must be upgraded before it is used with the 11.2 release. You must shutdown each database and take a complete back up before you perform an upgrade.

To associate Oracle Database 10.2 with the new 11.2 Oracle home, use `ORA_ROOT:[rdbms]migrate102.com`.

> **See Also:** *Oracle Database Upgrade Guide*

Note: HP OpenVMS does not support Database Upgrade Assistant utility.

4.3.2 Running the UT LRP.SQL Script

To run the `UTLRP.SQL` script:

1. Log in to the oracle Database account.

2. Set the default to the Oracle home directory and run the following command:

```
$ @ORAUSER sid
```

where *sid* is the database instance ID of the target database.

3. Start SQL*Plus as follows:

```
$ SQLPLUS "/ AS SYSDBA"
```

4. If necessary, start the database:

```
SQL> STARTUP
```

5. Run the UTLRP.SQL script as follows:

```
SQL> @ORA_ROOT:[RDBMS.ADMIN]UTLRP.SQL
```

4.4 Changing User Passwords

Oracle recommends that you change the password for system user such as SYS, SYSTEM and SYSMAN immediately after installation. To change a password:

1. Start SQL*Plus by using the following command:

```
$ SQLPLUS
```

2. Connect with the user name and password that you want to change by using the following command:

```
Enter user-name: user_name
Enter password: password
```

3. Change the password by using the following command:

```
PASSWORD mumble
Changing password for mumble
New password: password
Retype new password: password
```

4.5 Performing Configuration Tasks as the SYSTEM User

You must log in to a System Administrator account to perform the following configuration task.

Automating Database Startup and Shutdown (Optional)

You can configure the system to automatically start Oracle Database when the system starts (in your SYSTARTUP procedure) and to shut down Oracle Database when the system shuts down (in your SYSHUTDWN procedure). Automating database startup is optional, but automatic shutdown is recommended because it guards against improper shutdown of the database.

4.6 Performing Configuration Tasks as Oracle User

You must log in to the Oracle Database account to set initialization parameters.

Setting Initialization Parameters

When you create a database, the INIT*sid*.ORA parameters are automatically set. You can manually modify the initialization parameters in the init*sid*.ora with an HP OpenVMS editor. Activate the modified init*sid*.ora file by shutting down and restarting the database.

Do not use logical names in parameter files unless they are defined at an appropriate level (in other words, at a group level or the system level).

> **See Also:** *Oracle Database Administrator's Reference for HP OpenVMS Itanium* for information about `INIT.ORA` parameters

4.7 Required Product-Specific Postinstallation Tasks

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

- Section 4.7.1, "Configuring Oracle Net Services"
- Section 4.7.2, "Configuring Oracle Label Security"
- Section 4.7.3, "Installing Oracle Text Supplied Knowledge Bases"
- Section 4.7.4, "Configuring Oracle Messaging Gateway"
- Section 4.7.5, "Oracle Precompilers"

> **Note:** You must perform postinstallation tasks only for the products that you intend to use.

4.7.1 Configuring Oracle Net Services

If you have a previous release of Oracle software installed on this system, you may want to copy information from the Oracle Net Services `TNSNAMES.ORA` and `LISTENER.ORA` configuration files from the previous release to the corresponding files for the new release.These files should have a record format of `stream_lf`. If you copy a TNSNAMES.ORA or a LISTENER.ORA file from an earlier version, use the following command to convert the files:

```
$ CONVERT/FDL=SYS$INPUT filename
 filename
 RECORD
 CARRIAGE_CONTROL carriage_return
 FORMAT stream_LF ^Z
```

> **Note:** The default location for the `TNSNAMES.ORA` and `LISTENER.ORA` files is the `ORA_ROOT:[NETWORK.ADMIN]` directory.

Modifying the listener.ora File

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

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

Modifying the tnsnames.ora File

Unless you use a central `TNSNAMES.ORA` file, copy Oracle Net Services service names and connect descriptors from the previous release `TNSNAMES.ORA` file to the version of that file used by the new release.

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

4.7.2 Configuring Oracle Label Security

If you install Oracle Label Security, you must configure it in a database before you use it. You can configure Oracle Label Security in two ways, with or 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 the environment and Oracle Database must be registered in the directory.

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

4.7.3 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, you can install two supplied knowledge bases (English and French) from the Oracle Database 11*g* Release 2 (11.2) installation media.

Note: You can extend the supplied knowledge bases depending on your requirements. Alternatively, you can create knowledge bases, possibly in languages other than English and French. See *Oracle Text Reference* for more information about creating and extending knowledge bases.

4.7.4 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 the following lines to the `SID_LIST` parameter for the listener in the `LISTENER.ORA` file:

```
  (SID_DESC =
```

```
    (SID_NAME = mgwextproc)
    (ORACLE_HOME = oracle_home)
    (PROGRAM = extproc)
  )
```

For example, set the SID_LIST parameter as follows:

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PLSExtProc)
      (ORACLE_HOME = somedisk:[oracle.product.11_2_0.db_1])
      (PROGRAM = extproc)
    )
    (SID_DESC =
      (SID_NAME = mgwextproc)
        (ORACLE_HOME = somedisk:[oracle.product.11_2_0.db_1])
        (PROGRAM = extproc)
    )
  )
```

In this example:

- *ORACLE_HOME* is the path of the Oracle home directory.
- extproc is the external procedure agent executable file.

Modifying the TNSNAMES.ORA File for External Procedures

To modify the ORACLE_HOME:[NETWORK.ADMIN]TNSNAMES.ORA file for external procedures:

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

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

In this example:

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

Setting Up the MGW.ORA Initialization File

To modify the ORA_ROOT:[MGW.ADMIN]MGW.ORA file for external procedures, define the CLASSPATH logical name. A sample MGW.ORA file can be found in the ORA_ROOT:[MGW.ADMIN]SAMPLE_MGW.ORA directory.

You must include the classes in Table 4–1 and any additional classes required for Oracle Messaging Gateway to access non-Oracle messaging systems, such as WebSphere MQ or TIBCO Rendezvous classes.

Table 4–1 Classes for CLASSPATH Logical Name

Classes	Path
Oracle Messaging Gateway	ORA_ROOT:[MGW.JLIB]MGW.JAR
JRE run time	ORA_ROOT:[JRE.LIB]RT.JAR
Oracle JDBC	ORA_ROOT:[JDBC.LIB]OJDBC14.JAR
Oracle internationalization	ORA_ROOT:[NLSRTL.JLIB]ORAI18N.JAR
SQLJ	ORA_ROOT:[SQLJ.LIB]TRANSLATOR.JAR ORA_ROOT:[SQLJ.LIB]RUNTIME12.JAR
JMS Interface	ORA_ROOT:[RDBMS.JLIB]JMSCOMMON.JAR
Oracle JMS implementation	ORA_ROOT:[RDBMS.JLIB]AQAPIL3.JAR
Java Transaction API	ORA_ROOT:[JLIB]JTA.JAR

4.7.5 Oracle Precompilers

The following sections describe references and configuration files that are associated with Oracle precompilers.

Pro*C/C++

For the Pro*C/C++ precompiler, the configuration file is ORA_ROOT:[PRECOMP.ADMIN]PCSCFG.CFGS. This file installs without content. Use any text editor to customize it to the site requirements. For more information about configuring the pcscfg.cfg file, see *Pro*C/C++ Programmer's Guide*.

Pro*COBOL

For the Pro*COBOL precompiler, the configuration file is PCBCFG.CFG. This file installs without content. Use any text editor to customize it to the site requirements. For more information about configuring the pcbcfg.cfg file, see *Pro*COBOL Programmer's Guide*.

Pro*FORTRAN

For the Pro*FORTRAN precompiler, the configuration file is PCCFOR.CFG. This file installs without content. Use any text editor to customize it to the site requirements. For more information about configuring the pccfor.cfg file, see *Pro*FORTRAN Supplement to the Oracle Precompilers Guide*.

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:

5.1 Reviewing Accounts and Passwords

Note: Oracle Database 11*g* Release 2 passwords are case sensitive.

All databases created by 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.

Note: You can use the Oracle Enterprise Manager Database Control to view the complete list of database accounts. For more information, see section "Unlocking and Resetting User Passwords" on page 5-3.

Table 5–1 Database Accounts

User Name	Default Password	Description	See Also
ANONYMOUS	ANONYMOUS	Allows HTTP access to Oracle XML DB.	None
CTXSYS	CTXSYS	The Oracle Text account.	*Oracle Text Reference*
DBSNMP	Password set during installation or in the Database Configuration Assistant	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*
DMSYS	DMSYS	The data mining account. DMSYS performs data mining operations.	*Oracle Data Mining Administrator's Guide*
HR	HR	The account that owns the Human Resources schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	*Oracle Database Sample Schemas*
LBACSYS	LBACSYS	The Oracle Label Security administrator account.	*Oracle Label Security Administrator's Guide*
MDDATA	MDDATA	The schema used by Oracle Spatial for storing Geocoder and router data.	*Oracle Spatial Developer's Guide*
MDSYS	MDSYS	The Oracle Spatial and Oracle *inter*Media Locator administrator account.	*Oracle Spatial Developer's Guide*
OE	OE	The account that owns the Order Entry schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	*Oracle Database Sample Schemas*
ORDPLUGINS	ORDPLUGINS	The Oracle *inter*Media user. Plugins supplied by Oracle and third-party plugins are installed in this schema.	*Oracle Multimedia Reference*
ORDSYS	ORDSYS	The Oracle *inter*Media administrator account.	*Oracle Multimedia Reference*
OUTLN	OUTLN	The account that supports plan stability. Plan stability enables you to maintain the same execution plans for the same SQL statements. OUTLN acts as a role to centrally manage metadata associated with stored outlines.	*Oracle Database Concepts*
PM	PM	The account that owns the Product Media schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas.	*Oracle Database Sample Schemas*
SCOTT	TIGER	An account used by Oracle sample programs and examples.	*Oracle Database Administrator's Guide*
SH	SH	The account that owns the Sales History schema included in the Oracle Sample Schemas. It is available only if you loaded the Sample Schemas during an Enterprise Edition installation.	*Oracle Database Administrator's Guide*

Table 5–1 (Cont.) Database Accounts

User Name	Default Password	Description	See Also
SYS	Password set during installation or in the Database Configuration Assistant	The account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
SYSTEM	Password set during installation or in the Database Configuration Assistant	Another account used to perform database administration tasks.	*Oracle Database Administrator's Guide*
WKPROXY	WKSYS	The Ultra Search proxy user.	*Oracle Ultra Search Administrator's Guide*
XDB	XDB	The account used for storing Oracle XML DB data and metadata.	*Oracle XML DB Developer's Guide*

5.2 Unlocking and Resetting User Passwords

Passwords for all Oracle system administration accounts except `SYS`, `SYSTEM`, `SYSMAN`, and `DBSMP` are revoked after installation. Before you use a locked account, you must unlock it and reset its password. If you created a preconfigured database during the installation, but you did not unlock a required account, then you must unlock it, by using one of the following methods:

- Section 5.2.1, "Using Grid Control to Unlock Accounts and Reset Passwords"
- Section 5.2.2, "Using SQL*Plus to Unlock Accounts and Reset Passwords"

> **Note:** If you use Database Configuration Assistant to create a database. Click Password Management to unlock accounts after the database is created, before you exit from Database Configuration Assistant.

5.2.1 Using Grid Control to Unlock Accounts and Reset Passwords

To unlock and reset user account passwords using Oracle Enterprise Manager Grid Control, perform the following steps:

1. Log on to the Grid Control.
2. Click **Administration**.
3. In the Security section of the Administration screen, 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 has expired.
4. Select the user account that you want to modify, then click **Edit**.
5. Use the General screen of the Users property sheet to unlock the account and (optionally) to change the password.

See Also: Click **Help** in the Grid Control screen for more information about using the Grid Control

5.2.2 Using SQL*Plus to Unlock Accounts and Reset Passwords

Perform the following steps to unlock and reset user account passwords manually:

1. Start SQL*Plus and log in as `SYSDBA`:

```
$ SQLPLUS/NOLOG
SQL> CONNECT / AS SYSDBA
```

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

In this example:

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

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

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

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

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

5.3 Identifying Databases

The Oracle Database 11*g* Release 2 (11.2) 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:

```
host.example.com
```

In this example:

- `host` is the name of the database. The database name is a string of maximum eight characters that can contain alphanumeric, underscore (_), and hash (#) characters. The `DB_NAME` initialization parameter specifies the database name.
- `example.com` is the network domain in which the database is located. The database name and the network domain constitute the unique global database name. The domain is a string of maximum 128 characters that can contain alphanumeric,

underscore (_), and hash (#) characters. The DB_DOMAIN initialization parameter specifies the domain name.

The DB_NAME and DB_DOMAIN parameters combine to create the global database name value. This value is 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.

Note: The SID must contain no more than 6 characters on HP OpenVMS-based systems.

5.4 Locating the Parameter Files

The database initialization parameter file is generated when a database is created. By default, the preconfigured database uses a server parameter file named SPFILE*sid*.ORA. The following subsections describe the process of locating the parameter files:

- Section 5.4.1, "Locating the INIT.ORA File"
- Section 5.4.2, "Locating the Server Parameter File"

5.4.1 Locating the INIT.ORA File

The database initialization parameter file (INIT.ORA) is generated when a database is created. Use the INIT.ORA file to start the database instance, and create the database and data dictionary. When the database creation is complete, the instance is shutdown, and the INIT.ORA file is saved to ORA_ROOT:[DBS]INIT*sid*.ORA. The logical name ORA_PARAMS, points to ORA_ROOT:[DBS]INIT*sid*.ORA.

The INIT.ORA file is a text file that contains a list of preconfigured instance configuration parameters that must exist for an instance to start. You do not need to edit this file to use the starter database.

> **See Also:** *Oracle Database Administrator's Guide* and *Oracle Database Reference* for more information about database-specific initialization parameters and their default values

5.4.2 Locating the Server Parameter File

By default, the preconfigured database uses a server parameter file named SPFILE*sid*.ORA, which is stored in the ORA_ROOT:[DBS] directory.

If the server parameter file is not located in this directory, then the database uses the SPFILE parameter in an initialization parameter file to locate it. The default file name and location of this initialization parameter file is ORA_ROOT:[DBS]INIT*sid*.ORA.

You can use the Oracle Enterprise Manager to view the location of the server parameter file and list all the initialization parameters as follows:

1. Log on to the Enterprise Manager Control.
2. Under Targets tab, select the correct database target to display its home screen.
3. Click **Administration**.

4. In the Instance section of the Administration screen, click **All Initialization Parameters**.

   Enterprise Manager displays a table listing the current value of each initialization parameter.

5. Click **SPFile**.

   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 preceding the table.

5.5 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:

- Section 5.5.1, "Identifying Tablespaces and Data Files"
- Section 5.5.2, "Locating Redo Log Files"
- Section 5.5.3, "Locating Control Files"

5.5.1 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* Release 2 (11.2) databases.

Table 5–2 describes the tablespaces that are provided by the default preconfigured database and a description of what they store.

Table 5–2 Tablespaces and Descriptions

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

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

To use the Oracle Enterprise Manager Grid Control and to view the list of data files and their associated tablespaces:

1. Log on to the Enterprise Manager Control.
2. Under Targets tab, select the correct database target to display its home screen.
3. Click **Administration**.
4. In the Storage section of the Administration screen, click **Datafiles**.

   Enterprise Manager displays a table listing each data file and the tablespace with which it is associated.

Note: For more information about using the Grid Control to view, modify, and create tablespaces, click **Help** in the Grid Control screen.

5.5.2 Locating Redo Log Files

The starter database and the custom database each contain three redo log files that are located in the `ORA_ROOT:[ORADATA.db_name]` directory. The preconfigured database uses three redo log files. These files record all the changes made to data in the database buffer cache. If an instance fails, then Oracle Database uses the redo log files to recover the modified data in memory.

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

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

To use the Oracle Enterprise Manager Database Control to view or modify the redo log files for the preconfigured database, perform the following steps:

1. Log on to the Enterprise Manager Control.
2. Under the Targets tab, select the correct database target to display its home screen.
3. Click **Administration**.
4. In the Storage section of the Administration screen, click **Redo Log Groups**.

   Oracle Enterprise Manager displays a table listing the redo log groups that the database uses.
5. To view the name and location of the redo log file associated with a particular group, select that group and 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

5.5.3 Locating Control Files

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

The starter database contains three control files. These files are located in the `ORA_ROOT:[ORADATA.db_name]` directory. The `CONTROL101.CTL`, `CONTROL102.CTL`, and `CONTROL103.CTL` files are automatically included in the custom database. A preconfigured database uses three control files. Oracle recommends that you keep at least three control files (on separate physical drives) for each database and set the `CONTROL_FILES` initialization parameter to list each control file.

To use the Oracle Enterprise Manager Grid Control to view information about the control files for the preconfigured database, perform the following steps:

1. Log on to the Enterprise Manager Control.
2. Under Targets tab, select the correct database target to display its home screen.
3. Click **Administration**.
4. In the Storage section of the Administration screen, click **Controlfiles**.

   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, see *Oracle Database Administrator's Guide*.

5.6 Adding or Upgrading Products

The following section describe the steps to add or upgrade Oracle products:

- Section 5.6.1, "Preparing to Add a Product"
- Section 5.6.2, "Adding a Product Summary"

5.6.1 Preparing to Add a Product

Perform the following steps before you add or upgrade a product:

1. Verify that the system has Oracle Database installed.
2. Log on to the computer.
3. Run the `ORAUSER` command procedure for the database:

   ```
   $ SET DEFAULT disk:[oracle_home_path]
   $ @ORAUSER sid
   ```

4. If you plan to upgrade any Oracle product tables, then change the Oracle `SYSTEM` password to `MANAGER`.

Note: If you do not change the Oracle SYSTEM password to MANAGER, then the installation may fail. Oracle installation scripts often assume that the password is MANAGER. Alternatively, you can check for hard-coded user names and passwords in Oracle scripts and change them to the required user names and passwords.

5. Back up the old code and database files (data, redo log, and control).
6. Mount the distribution media.
7. Run Oracle Universal Installer, and select **Custom install**.

After completing these preliminary steps, you can load and configure Oracle products.

5.6.2 Adding a Product Summary

To add a product summary:

1. Start Oracle Universal Installer as follows:

```
$ SET DEFAULT dvd_device:[DATABASE]
$ @RUNINSTALLER.COM
```

2. Point to the existing Oracle home.
3. Select **Custom Install**.
4. Select **Products to Install**.
5. Configure the products if Oracle Universal Installer asks for any specifications.
6. Click **Install**.
7. Exit Oracle Universal Installer.
8. Start an instance.
9. Build or upgrade the database tables menu.

5.7 Relinking the Oracle Code

The Oracle code must be relinked under the following conditions:

- Change of operating system (including an upgrade).
- Changes made to Oracle Net Services configuration.
- Changes made to Oracle RDBMS configuration.

The following sections describe how you can relink Oracle code.

Step 1 Shutting Down the Given Databases

Verify that you point to the right database instance. To perform this task, enter the following at the DCL prompt:

```
$ SHOW LOGICAL ORA_SID
ORA_SID" = "V11G" (LNM$JOB_TABLE)
```

Start SQL*Plus and connect to SYS. Check the V$PROCESS view to see if other processes are active.If no other processes are active, and if you are pointing to the correct instance, then shut down the given database.

Oracle background processes stop after the database is shut down. To verify, enter the `SHOW SYSTEM` command at the DCL prompt.

Step 2 Removing Images

To remove images, enter the following command at the system prompt:

```
$ REMORACLE
```

> **Note:** Generally, no response is returned except for the return of the DCL prompt.

Step 3 Relinking the Oracle Code

To relink the Oracle code, enter the following commands at the DCL prompt:

```
@ORA_ROOT:[BIN]RELINK product | ALL
@ORA_ROOT:[BIN]RELINK DEPLOY
INSORACLE
```

6

Removing Oracle Software

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

- Section 6.1, "Identifying All Instances"
- Section 6.2, "Removing an Oracle Database"
- Section 6.3, "Removing Oracle Software"

> **Note:** If you want to remove an Oracle Real Application Clusters (Oracle RAC) installation, then see *Oracle Real Application Clusters Installation and Configuration Guide* for more information.
>
> If you want to remove an individual product, then check the product-specific documentation for requirements and restrictions.

6.1 Identifying All Instances

To identify all instances associated with the Oracle home that you want to remove, log in to the Oracle Database account and enter the following command:

```
$ SHOW LOGICAL SYS$LOGIN
```

Example output:

```
DISK$ORACLE1:[ORACLE11G]
```

The instance names are in the `ORATAB` file in the subdirectory for the current node:

```
$ TYPE DISK$ORACLE1:[ORACLE11G.nodename]ORATAB.
```

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

```
$ORACLE_SID:ORACLE_HOME:Y/N
```

Where:

`SID` is the system identifier of the Database.

`ORACLE_HOME` is the home directory of the Database.

For Example:

```
SID:/VMSDISK/ORACLE_HOME:N
```

6.2 Removing an Oracle Database

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

> **Caution:** Removing an Oracle Database deletes all of the data in the database. If you want to keep this data, then ensure that you back up the database before deleting it.

1. Log in to the Oracle Database account.
2. Enter the following command to change the directory to the default database location:

   ```
   $ set default oracle_home_directory
   ```

3. Run the `ORAUSER` command procedure, specifying the `SID` for the database that you want to remove:

   ```
   $ @ORAUSER sid
   ```

4. Use the `$ SET DISPLAY` command to direct X-Windows to your PC or workstation.
5. Start Database Configuration Assistant by using the following command:

   ```
   $ DBCA
   ```

   The Welcome screen is displayed.
6. Click **Next**.

   The Operations screen is displayed.
7. Select **Delete a Database**, then click **Next**.
8. Select the database that you want to delete, and then click **Finish**.
9. In the screen that is displayed, confirm that you want to delete the database.
10. When Database Configuration Assistant removes the database, you are prompted to select 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.

6.3 Removing Oracle Software

The following steps describe how to use the 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. Remove any database instances that are associated with this Oracle home, and delete or reconfigure the Oracle Cluster Synchronization Services daemon.
2. Stop any processes running in this Oracle home:

Table 6–1 Commands to Stop Processes Running in Oracle Home

Process Name	Command
Database Control	`$ EMCTL STOP AGENT`
Oracle Net Services listener	`$ LSNRCTL STOP`

3. If necessary, log in to the Oracle Database account.
4. Run the `ORAUSER.COM` file to redefine the Oracle home.

   ```
   $ @ORAUSER
   ```

5. Use the `$ SET DISPLAY` command to direct X-Windows to your PC or workstation.
6. Start Oracle Universal Installer as follows:

   ```
   $ @dvd_device:[DATABASE]RUNINSTALLER
   ```

   If the product was installed from the `[CLIENT]` or `[COMPANION]` directory of the Installation Media, then use that directory instead of `[DATABASE]`.
7. In the Welcome screen, click **Deinstall Products**.

   The Inventory screen is displayed, listing all of the Oracle homes on the system.
8. In the Inventory screen, select the Oracle home and the products that you want to remove, then click **Remove**.

   Caution: If you select to remove Oracle JVM, then Oracle Universal Installer removes all installed products that depend on Oracle JVM, including Oracle Database 11*g* Release 2 (11.2).

9. When Oracle Universal Installer displays a confirmation screen asking you to confirm that you want to uninstall the products and their dependent components, click **Yes**.

   The Installer displays a progress indicator as it removes the software.
10. When the products have been deleted, click **Cancel** to exit from Oracle Universal Installer, then click **Yes**.

    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.

A

Oracle Database 64-Bit Feature

This appendix introduces the 64-bit feature or Very Large Memory (VLM) and recommends the parameter settings for this feature.

The topics covered in this appendix are:

- Section A.1, "Introduction to Oracle Database 64-Bit Feature"
- Section A.2, "Suggested Parameter Settings"
- Section A.3, "Other Recommendations"

> **Note:** The 64-bit feature is a standard feature of Oracle Database for HP OpenVMS and cannot be removed. The 64-bit feature tuning is under constant revision. Therefore, this information will be updated as further tuning suggestions are realized.

A.1 Introduction to Oracle Database 64-Bit Feature

The Oracle 64-bit feature provides the ability to support VLM system configurations with large amounts of RAM.

The following sections describe the benefits of the 64-bit feature and its implementation:

- Section A.1.1, "Benefits of Oracle Database 64-Bit Feature"
- Section A.1.2, "Implementation of Oracle 64-Bit Feature"

A.1.1 Benefits of Oracle Database 64-Bit Feature

The primary benefit of the 64-bit feature is performance, because many operations can now run at memory speed instead of disk speed. With larger amounts of data in the memory, the database issues fewer calls to disk. This greatly reduces the delay of disk I/O.

Oracle Database 64-bit feature also provides the advantage of scalability to support more users and larger volumes of data. The system does not need to swap data in and out of memory to process all the transaction requests, and it can effectively accommodate requests for large amounts of data.

The 64-bit feature benefits both query-intensive decision support system (DSS) and read/write (OLTP) transactions. The 64-bit feature provides index builds, full table scans, ad hoc queries, and multiway joins for DSS. For OLTP, the feature provides the ability to support very large tables, large amounts of data, and large number of users.

A.1.2 Implementation of Oracle 64-Bit Feature

The HP OpenVMS operating system has native 64-bit memory addressing, which enables Oracle Database to implement the 64-bit feature. The need to estimate the maximum System Global Area (SGA) at installation time is now eliminated. This speeds up process startup. The server no longer includes an SGAPAD.

By default, SGA creation uses HP OpenVMS facilities to create global sections.

A.2 Suggested Parameter Settings

The recommended parameter settings for the Oracle 64-bit feature are:

1. Big Oracle Blocks (BOB) provide the ability to support larger I/O transfers between memory and disk. BOB complements large SGA configurations, because BOB enables the system to move data faster between memory and disk. With VLM configurations, system performance depends directly on the ability of the system to move database blocks into the SGA as efficiently as possible. Without the benefits of improved data transfer, performance can decline.

   For a pure DSS application, select a large value (such as 32 KB) for DB_BLOCK_SIZE. For an OLTP type of application, choose a lower value (such as, 2 KB or 4 KB). The larger the DB_BLOCK_SIZE, the more serious the impact on single-row lookups.

2. Configure the size of the Oracle buffer cache such that it provides the best possible cache hit ratio without affecting memory requirements of other Oracle and system processes.

   Note: You cannot combine the DB_BLOCK_BUFFERS with the dynamic DB_CACHE_SIZE parameter. If you combine these parameters in the same parameter file, then it will produce an error.

   For example, for a 3 GB buffer cache with DB_BLOCK_SIZE=8192, set DB_BLOCK_BUFFERS=400000.

3. Set the SORT_AREA_SIZE parameter with care. SORT_AREA_SIZE is the space used in Program Global Area (PGA) for each sort run by each Oracle process. If the value is too high, then the PGA will use excessive memory when sorting. The default value of 512 KB is usually sufficient.

   Check statistics, such as V$SYSSTAT, to see if the number of sorts to disk is high compared to in-memory sorts. If it is, then increase the value of SORT_AREA_SIZE.

A.3 Other Recommendations

Check the size of the number of rows of the tables involved in the query, and translate this size into the total number of blocks. Based on the query, try to fit as many of the frequently accessed table blocks in the buffer cache.

For example, if four tables are involved in a query, but columns from one table are used repeatedly in the WHERE clause in joins, IN, and so forth, then try to fit as many blocks as possible from this table into the cache to see if DB_BLOCK_BUFFERS can be increased. To ensure that the frequently accessed tables are cached and stay in the most recently used (MRU) end of the cache, perform either of the following steps:

- Enter the following command (using SQL*Plus):

  ```
  alter table tablename cache
  ```

or

- At the time of table creation, enter the following:

```
create table tablename ... cache
```

If an adequate number of buffers are available to accommodate all blocks from all tables that are involved in the query, then use the `alter` command to cache all the blocks. The purpose is to cache most blocks into memory to ensure that I/O to disks is eliminated or remains low.

B

Noninteractive Installation and Configuration

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

- Section B.1, "Silent Mode Installation"
- Section B.2, "Performing Silent Installation"
- Section B.3, "Running Oracle Universal Installer in Silent Mode or Suppressed Mode"
- Section B.4, "Running Oracle Network Configuration Assistant in Silent Mode"
- Section B.5, "Running Database Configuration Assistant in Noninteractive or Silent Mode"

B.1 Silent Mode Installation

A silent installation with Oracle Universal Installer is noninteractive, meaning that it takes place without using the Java graphical user interface (GUI). Oracle Universal Installer gets its information from a text file that is specified at the command line by using the `-RESPONSEFILE` argument instead of interactive GUI input and output. If the `-RESPONSEFILE` flag is given, then the installation is assumed to be a silent mode type.

> **Note:** Because the design of Oracle Universal Installer causes GUI classes to be loaded (but not displayed), a silent installation still requires an X-Windows connection.

A silent installation can also be performed in `BATCH` mode, if the process has established an X Windows connection through the `SET DISPLAY` command.

This section discusses the following topics:

- Section B.1.1, "What Is a Response File?"
- Section B.1.2, "Why Perform a Silent Installation?"
- Section B.1.3, "Modifying a Response File"

B.1.1 What Is a Response File?

A response file contains answers to installation questions that otherwise would be provided by the user in an interactive installation session. Each answer is stored as a value for a variable that is identified in the response file. For example, values for `ORACLE_HOME` or `Install Type` can be set automatically within the response file. Response file templates are generated by the installation developer. The response file

template for the installation of the product can be found on the installation DVD under the [DATABASE.RESPONSE] directory. You should be able to find one for your choice of install type (for example, ENTERPRISE.RSP for an Enterprise Edition server installation, or CUSTOM.RSP for a custom installation).

B.1.2 Why Perform a Silent Installation?

Silent installations can be useful to install an Oracle product multiple times on multiple computers, or if you prefer not to use the GUI because of performance reasons over a slow remote X Windows connection. If the options that you select while installing on each computer are always the same, then you save the time of reviewing each installation screen and selecting the various installation options.

Silent installations can also ensure that multiple users in your organization use the same installation options when they install Oracle products. You can therefore more easily support those users because you already know what components and options have been installed on each computer.

Before you perform a silent installation, you should review the settings in the response file template that is provided with the Oracle product.

B.1.3 Modifying a Response File

If the product installation includes a response file template, then you can find it on the shipped media in the *device*:[DISK1.RESPONSE] directory. See Section B.1.3.1, "List of Response File Names" for a list of response files,. Mandatory items have the following comment just before the definition:

```
# OPENVMS_VALUE_REQUIRED
```

Three values must be provided:

- FROM_LOCATION
- ORACLE_HOME
- ORACLE_HOME_NAME

FROM_LOCATION is the complete path of the source of the products to be installed (in other words, to the PRODUCTS.XML file in the STAGE subdirectory of the DATABASE, CLIENT, COMPANION, or CLUSTERWARE main directory of the DVD) from the shipped media. If contents of shipped media were copied to a local device, then specify that path. ORACLE_HOME is the directory into which the Oracle software will be installed. The account from which Oracle Universal Installer is run must have appropriate privileges to create a root-level directory if the directory that was specified for ORACLE_HOME does not already exist.

Note:

- FROM_LOCATION must be in UNIX-style syntax.
- ORACLE_HOME must be on an ODS-5 disk.

To modify the response file:

1. Make a copy of the response file for the product, and open it in a text editor.
2. Search for the three items that are listed in the preceding paragraph (or for the string OPENVMS_VALUE_REQUIRED), and provide values for those items.

3. Modify other fields as applicable (see comments within the response file that you are modifying).

B.1.3.1 List of Response File Names

The following table lists the response file names and their descriptions:

File Name	Description
ENTERPRISE.RSP:	Enterprise Edition Installation of Oracle Database Server
CUSTOM.RSP	Custom Edition Installation of Oracle Database Server
CLIENTADMIN.RSP	Administrator installation of Oracle Database Client
CLIENTRUNTIME.RSP	Run time installation of Oracle Database Client
CLIENTCUSTOM.RSP	Custom installation of Oracle Database Client
DBCA.RSP	Database Configuration Assistant

B.2 Performing Silent Installation

You can install Oracle software noninteractively by specifying a response file when you start Oracle Universal Installer. The Installer uses the values contained in the response file to provide answers to some or all of Oracle Universal Installer prompts. If you include responses for all of Oracle Universal Installer prompts in the response file, then you can run a completely noninteractive installation.

To install and configure Oracle products noninteractively, perform the following steps:

- Section B.2.1, "Creating the ORAINST.LOC File"
- Section B.2.2, "Preparing Response Files"
- Section B.2.3, "Running Oracle Universal Installer Noninteractively"

These steps are described in the following sections.

B.2.1 Creating the ORAINST.LOC File

If you plan to install Oracle products noninteractively, 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 that are installed on the system.

> **Note:** If Oracle software has been installed previously on the system, then the ORAINST.LOC file may already exist. If the file does exist, then you do not need to create a new file.

To create the ORAINST.LOC file, perform the following steps:

1. Set default directory to the Oracle oraInventory directory.The ORAINST.LOC file resides in a subdirectory under the SYS$LOGIN

   ```
   $ SET DEFAULT logindisk:[logindir.nodename]
   ```

2. Use a text editor to create the ORAINST.LOC file, containing the following lines:

   ```
   inventory_loc=oraInventory_location
   inst_group=group_id
   ```

In this example, *oraInventory_location* is the location of the Oracle Inventory directory which is on an ODS-5 volume, and *group_id* is the group ID of the Oracle Database account.

3. Set the file protection on the ORAINST.LOC file as follows:

```
$ SET FILE/PROT=(O:RWED,S:RWED,:rwe,w:re):oraInst.loc
```

B.2.2 Preparing Response Files

Oracle provides response file templates for each product and installation type and for each configuration tool. These files are located in the [DATABASE.RESPONSE] directory on the installation media.

Table B–1 lists the response files that are provided with Oracle Database 11*g* Release 2 (11.2).

Table B–1 Response Files

Response File	Description
ENTERPRISE.RSP	Enterprise Edition installation of Oracle Database.
CUSTOM.RSP	Custom installation of Oracle Database.
DBCA.RSP	Database Configuration Assistant.
NETCA.RSP	Oracle Net Services Configuration Assistant.

To prepare a response file:

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

```
$ COPY dvd_device:[DB.RESPONSE]response_file dest_path
```

In this example, *src_path* is the installation media mount point directory or the db directory on the DVD. If you have copied the software to a hard drive, then you can edit the file in the RESPONSE directory if you prefer.

2. Open the response file in a text editor:

```
$ EDIT [dest_path]response_file
```

3. Edit the file by following the directions in Section B.1.3, "Modifying a Response File" on page B-2.

> **Note:** Oracle Universal Installer or the configuration assistant fails if you do not correctly configure the response file. See Section F.6, "Silent Response File Error Handling" for more information about troubleshooting a failed noninteractive installation.

B.2.3 Running Oracle Universal Installer Noninteractively

To run Oracle Universal Installer noninteractively, enter a command similar to the following:

```
$ @[directory_path]RUNINSTALLER [-SILENT] -RESPONSEFILE filename
```

In this command:

- *directory_path* is the DATABASE, CLIENT, COMPANION, or CLUSTERWARE directory in the hard drive copy of the DVD
- -SILENT specifies a completely noninteractive installation
- *filename* is the full path and file name of the installation response file that you configured

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

B.3 Running Oracle Universal Installer in Silent Mode or Suppressed Mode

To run Oracle Universal Installer in silent mode or suppressed mode, use the following steps:

1. Perform the preinstallation tasks listed in Chapter 2, "Preinstallation Tasks".
2. Log in to the Oracle Database account (typically, oracle).
3. If you are performing a suppressed-mode installation, set the DISPLAY.
4. To start Oracle Universal Installer in silent mode or suppressed mode, enter a command similar to the following command:

```
$ @[directory_path]RUNINSTALLER [-SILENT] -RESPONSEFILE filename
```

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

In the example:

- *directory_path* is the DATABASE, CLIENT, COMPANION, or CLUSTERWARE directory in the hard drive copy of the DVD.
- -SILENT indicates that you want to run Oracle Universal Installer in silent mode.
- *filename* is the full path and file name of the installation response file that you configured.

> **Note:** For more information about other options for the RUNINSTALLER command, enter the following command:
>
> ```
> $ @[directory_path]RUNINSTALLER -HELP
> ```

B.4 Running Oracle Network Configuration Assistant in Silent Mode

You can run Oracle Network Configuration Assistant in silent mode to configure and start an Oracle Net Services listener on the system, to configure naming methods, and to configure Oracle Net Services service names. To run the Oracle Network 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 [DATABASE.RESPONSE] directory on the DVD.

To run Oracle Network Configuration Assistant in silent mode, perform the following steps:

1. Copy the `NETCA.RSP` response file template from the response file directory to a directory on the system.

   ```
   $ COPY [DATABASE.RESPONSE]NETCA.RSP local_directory
   ```

2. Open the response file in a text editor. For example:

   ```
   $ EDIT [local_directory]NETCA.RSP
   ```

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

   Note: Oracle Network Configuration Assistant stops if you do not correctly configure the response file.

4. Log in to the Oracle Database account (typically `oracle`) and set your Oracle environment by running `ORAUSER.COM` in the top-level Oracle home directory.
5. Enter a command similar to the following to run the Oracle Network Configuration Assistant in silent mode:

   ```
   $ @ORA_ROOT:[BIN]NETCA -SILENT -RESPONSEFILE=[local_dir]NETCA.RSP
   ```

   In this example:

   - `/silent` option indicates that you want to run the Oracle Network Configuration Assistant in silent mode.
   - *`local_dir`* is the full path of the directory where you copied the `NETCA.RSP` response file template.

B.5 Running Database Configuration Assistant in Noninteractive or Silent Mode

You can run Database Configuration Assistant in noninteractive or silent mode to configure and start 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.RESP` in the `[.db.response]` directory on the DVD.

Silent mode or noninteractive mode requires an appropriate response file. The following sections discuss how to use and run Database Configuration Assistant in silent mode or noninteractive mode:

- Section B.5.1, "Using Database Configuration Assistant in Noninteractive Mode"
- Section B.5.2, "Using Database Configuration Assistant in Silent Mode"
- Section B.5.3, "Running Database Configuration Assistant in Noninteractive or Silent Mode"

B.5.1 Using Database Configuration Assistant in Noninteractive Mode

In noninteractive mode, Database Configuration Assistant uses values that you specify in the response file or as command line options to create a database. As Database Configuration Assistant configures and starts the database, it displays a screen that contains status messages and a progress bar. The screen that it displays is the same

screen that is displayed when you create a preconfigured database during an Enterprise Edition installation.

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

B.5.2 Using Database Configuration Assistant in Silent Mode

In silent mode, Database Configuration Assistant uses values that you specify in the response file or as command-line options to create a database. Database Configuration Assistant does not display any screens or information while it creates the database. It logs all messages (including information, errors, and warnings) in a log file.

From the command line, enter the following command to see all of Database Configuration Assistant options that are available when you use the silent mode:

```
$ DBCA -help
```

B.5.3 Running Database Configuration Assistant in Noninteractive or Silent Mode

To run Database Configuration Assistant in noninteractive or silent mode, use the following steps:

> **Note:** Because Database Configuration Assistant design causes GUI classes to be loaded (even in silent mode), a silent installation requires an X-Windows connection.

1. Copy the `DBCA.RSP` response file template from the response file directory to a directory on the system:

   ```
   $ COPY device:[DB.RESPONSE]DBCA.RSP dest_path
   ```

   In this example, *`src_path`* is the installation media directory or the `db` directory on the DVD. If you have copied the software to a hard drive, then you can edit the file in the `[.DB.RESPONSE]` directory if you prefer.

2. Open the response file in a text editor as follows:

   ```
   $ EDIT NETCA.RSP
   ```

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

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

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 supported options, enter the following command:

```
$ DBCA -HELP
```

Because of the DCL limit of eight command-line parameters on HP OpenVMS, it is not possible to specify all the required parameters to create a database directly at the command line. Therefore, as a workaround, Database Configuration Assistant command-line arguments may be placed in a file called SYS$SCRATCH:DBCA.ARGS. The dbca.com script looks for this file, and if present, passes the specified arguments to Database Configuration Assistant Java application. The DBCA.COM command procedure ignores all arguments that are specified at the command line except: -SILENT, -RESPONSEFILE *response_file*, and -HELP. To avoid inadvertently applying the same arguments to the next invocation of Database Configuration Assistant when DBCA.ARGS exits, if an arguments file was used, then it is renamed to SYS$SCRATCH:DBCA.ARGS_OLD.

4. Log in to the Oracle Database account (typically oracle) and set the ORACLE_HOME logical name to specify the correct Oracle home directory.
5. If you intend to run Database Configuration Assistant in noninteractive mode, then set the DISPLAY.
6. Enter a command similar to the following to run Database Configuration Assistant in noninteractive or silent mode with a response file:

   ```
   $ DBCA -SILENT -RESPONSEFILE
   response_file
   ```

   In the example:

   - -silent indicates that you want to run Database Configuration Assistant in silent mode.
   - *response_file* is the full path to the copy of DBCA.RSP response file template that will be used.

The following sections contain examples that illustrate the use of silent mode:

Database Configuration Assistant Silent Mode Example 1: Creating a Clone Database

To create a clone database, enter the following in SYS$SCRATCH:DBCA.ARGS:

```
-silent
-createDatabase
-templateName Transaction_Processing.dbc
-gdbname ora11
-sid ora11
-datafileJarLocation /disk$disk1/oracle11g/ora11/assistants/dbca/templates
-datafileDestination /disk$disk1/oracle11g/ora11/oradata
-responseFile NO_VALUE
```

```
-characterset WE8ISO8859P1
```

> **Note:** File specifications in an argument file may be in either HP OpenVMS syntax or in UNIX syntax.

Then, ensure that the ORACLE_HOME environment is set up by running:

```
$ @disk:[oracle-home-path]ORAUSER
```

For example:

```
$ @DISK$DISK1:[ORACLE11G.ORA11]ORAUSER
```

Next, set the DISPLAY and start Database Configuration Assistant. Log files are present in the same location as with an interactive install. In this example:

```
DISK$DISK1:[ORACLE11G.ORA11.ADMIN.ORA11.CREATE]
```

In addition, check the ORA_ROOT:[ASSISTANTS.DBCA.LOGS] directory.

Database Configuration Assistant Silent Mode Example 2: Creating a Seed Template

To create a seed template, enter the following in SYS$SCRATCH:DBCA.ARGS:

```
-silent
-createCloneTemplate
-sourceDB ora11
-sysDBAUserName sys
-sysDBAPassword change_on_install
-templateName copy_of_ora11
-datafileJarLocation /disk$disk1/oracle11g/ora11/assistants/dbca/templates
```

Then, ensure that the oracle environment is set up by running the ORAUSER script, set the DISPLAY and start Database Configuration Assistant. Log files for creating a seed template are located in the ORA_ROOT:[ASSISTANTS.DBCA.LOGS] directory.

C

Installing, Configuring, and Running Enterprise Manager Agent

This appendix describes Oracle Enterprise Manager Agent for Oracle Database 11*g* Release 2 (11.2). It includes information about the following topics:

C.1 Introduction to Enterprise Manager Framework

This release of Enterprise Manager Agent is different from the earlier releases of Oracle Intelligent Agent in terms of architecture and implementation. Enterprise Manager Agent is part of the Enterprise Manager Framework. The Database Control and Application Control components are not supported on Oracle Database 11*g* Release 2 (11.2) for HP OpenVMS. To use Enterprise Manager Agent for HP OpenVMS, it is necessary to install the Oracle Management Server (OMS) component of Grid Control Enterprise Manager on a UNIX-based or Microsoft Windows-based computer. Enterprise Manager Agent running on HP OpenVMS will register with and function in coordination with the OMS.

The Enterprise Manager Agent in 11*g* Release 2 (11.2) will be shipped separately. However, customers may still use the Enterprise Manager Agent 10*g* Release 2 (10.2.0.2) kit. For more information on obtaining the Enterprise Manager Agent 10*g* Release 2 (10.2.0.2) kit, see the Release notes.

Enterprise Manager Agent is implemented in C, Java, and Perl. Unlike earlier versions, TCL scripts are no longer used. Perl scripts replaces the TCL scripts. Perl version 5.8 is included in the kit along with the Enterprise Manager Agent. Oracle Universal Installer automatically installs Perl. It is not required to install a separate Perl kit. This kit includes Oracle-specific DBD or DBI, and is therefore, the only supported Perl kit for use with Oracle Database. The logical name `PERL_ROOT` will be defined to the physical path `ORA_ROOT:[PERL]` as a job level logical name. To avoid potential conflict with Perl from any other kit, including any future versions supplied by Oracle, it is recommended not to convert this logical name to a `GROUP` or `SYSTEM` logical name.

The Job model supported in earlier releases is still available in this release in a similar, but not identical, fashion. Jobs can be submitted, their status can be monitored, and

their output checked. The Event model, as it existed in earlier versions, is no longer available. Instead, what is available is Targets. See Section C.5, "Supported Targets and Jobs" for more information.

> **See Also:** For more details, see the `README_VMS_11GR2.TXT` file that ships with the Oracle Database Server Kit. The information in this file supersedes the information in this appendix.

C.2 Installation Requirements

If you choose not to create a dedicated batch queue `ORA_BATCH_QUEUE_`*`hostname`* for Enterprise Manager Agent, then you must define a logical name `ORA_BATCH_QUEUE_`*`hostname`*(a Process or Job logical is sufficient) to the name of the batch queue to be used for running Enterprise Manager Agent. In addition to running in the batch queue, Enterprise Manager Agent also submits several other tasks, typically related to actions required on other Oracle installations on the same host.

The *`hostname`* part of the logical name should be the same as the value of the `TCPIP$INET_HOST` logical name. The batch queue specified should run at the same priority level as all other general purpose processes on the system, typically 4.

For example, the following qualifiers would be typical for creating the batch queue that is used for Enterprise Manager Agent.:

```
/BASE_PRIORITY=4 /CPUDEFAULT=INFINITE /CPUMAXIMUM=INFINITE /JOB_LIMIT=100
/OWNER=[SYSTEM] /PROTECTION=(S:M,O:RSD,G:S,W:S)
```

Enterprise Manager Agent internally spawns several subprocesses and detached processes to run most of its tasks. The type of actions performed in these processes ranges widely from running simple DCL commands to running Java applications. It is imperative that the account used to install and run Enterprise Manager Agent not modify the typical HP OpenVMS session environment in any way that would alter the expected output of normal DCL commands. The `LOGIN.COM` file of the account used to start or stop the agent should not have any such redefinitions. If the redefinitions are required for any reason, then it is recommended that they be disabled in the batch mode, because Enterprise Manager Agent runs in the batch mode.

C.3 Installation and Configuration

When you initially run Oracle Universal Installer, it instantiates certain files under the `ORA_ROOT:[SYSMAN.EMD]` and `ORA_ROOT:[SYSMAN.CONFIG]` directories. These are the Targets files and the Properties files. However, these cannot be used directly. Enterprise Manager Agent is configured correctly when Database Configuration Assistant is run. The configuration files are created under a separate directory structure (`ORA_ROOT:[`*`hostname_sid`*`.SYSMAN...]`). Because the DB Console is not supported on HP OpenVMS, it is necessary to ensure manual changes to the configuration file or files before starting Enterprise Manager Agent.

Manual Changes Required Before Attempting to Start Enterprise Manager Agent

You must manually change the following properties in the `ORA_ROOT:[`*`Hostname_SID`*`.SYSMAN.CONFIG]EMD.PROPERTIES` file before starting Enterprise Manager Agent.

1. `REPOSITORY_URL`

   This property is required to allow Enterprise Manager Agent to upload data to a central Grid Control Enterprise Manager Repository. When Enterprise Manager Agent starts, it registers its targets and other host configuration information with

the Oracle Management Server (OMS) so that they are displayed on the Grid Control Enterprise Manager Console. It is then possible to manage the targets from the Enterprise Manager Console. A sample `REPOSITORY_URL` property is as follows:

```
REPOSITORY_URL=http://orclhost01.example.com:1234/em/upload
```

2. `agentTZRegion`

   This property indicates the time zone in which Enterprise Manager Agent is running. To update this property in the `EMD.PROPERTIES` file, run the following command:

   ```
   $ EMCTL CONFIG AGENT UPDATETZ
   ```

   A sample `agentTZRegion` property for Pacific Standard Time is as follows:

   ```
   agentTZRegion=-08:00
   ```

C.4 Management and Maintenance

This section describes the procedures to manage and maintain Enterprise Manager Agent.

- Section C.4.1, "Startup, Shutdown, and Status"
- Section C.4.2, "Troubleshooting and Maintenance"

C.4.1 Startup, Shutdown, and Status

Perform the following steps to start, shut down, or view the status of Enterprise Manager Agent:

- After the manual configuration is complete, run the following command to start Enterprise Manager Agent:

  ```
  $ EMCTL START AGENT
  ```

- To shut down Enterprise Manager Agent, run the following command:

  ```
  $ EMCTL STOP AGENT
  ```

- To query the status of Enterprise Manager Agent at any time, run the following command:

  ```
  $ EMCTL STATUS AGENT
  ```

> **Note:** Enterprise Manager Agent is the only mechanism of communication between the Oracle Management Server and targets or applications running on the host. Therefore, it is not possible to start or shut down Enterprise Manager Agent from the Enterprise Manager Console.

C.4.2 Troubleshooting and Maintenance

Perform the following tasks to manage or troubleshoot Enterprise Manager Agent:

- Section C.4.2.1, "Enterprise Manager Agent Fails to Start"
- Section C.4.2.2, "Extended File Specification (EFS) Characteristics"
- Section C.4.2.3, "TMP Directory"

C.4.2.1 Enterprise Manager Agent Fails to Start

If Enterprise Manager Agent fails to start, check the following files for typical error messages:

Batch Job Log File

Enterprise Manager Agent is submitted as a batch job into the agent batch queue whenever it is started. Each time Enterprise Manager Agent starts, a fresh batch log file is created, which is available at `ORA_ROOT:[SYSMAN.LOG]START_AGENT_`*`HOST`*`.LOG`. Refer to the latest version of this log file. If it is readable (not locked by a running Enterprise Manager Agent process), look through the file to see if there are any failure messages. Typical errors would be `Failure to launch the Enterprise Manager Agent` because of some issues with related shared libraries.

Enterprise Manager Agent Log and Trace Files

If there are no errors in `ORA_ROOT:[SYSMAN.LOG]START_AGENT_`*`HOST`*`.LOG`, look at the `Enterprise Manager Agent.trc` and `Enterprise Manager Agent.log` files. These are located in the `ORA_ROOT:[`*`host_sid`*`.SYSMAN.LOG]` directory. A common error is `Address already in use`, when the port number on which Enterprise Manager Agent is listening is being used by some other application.

Note: If you are not able to restart the Agent, ensure that there are no processes on the system that start with `ORA_EM`*.

You can stop all the process by entering the command `$ STOP/ID=<process-id>` on each one of them and then try to restart the agent.

C.4.2.2 Extended File Specification (EFS) Characteristics

Enterprise Manager Agent requires an extended file-specifications environment to handle files with multiple dots in their names, long file names, and so on. The default Oracle environment (after running `ORAUSER.COM`) does not provide this environment. Internally, Enterprise Manager Agent tools set up this extended environment when started, and reset the environment back to the original when completed. In a normal run, it is not necessary for an Enterprise Manager Agent administrator to require this environment for interactive use. However, in certain situations, when there are failures due to interruption in the host system, or due to lack of resources, it may be required to manually manipulate certain files, which in turn, would need the EFS environment.

There are two scripts included with Enterprise Manager Agent kit that provides the EFS environment:

- `ORA_ROOT:[SYSMAN.BIN]EFS.COM`

  Enables the EFS environment.

- `ORA_ROOT:[SYSMAN.BIN]NOEFS.COM`

  Resets the environment to non-EFS.

For convenience, two DCL symbols have been created to set and reset the EFS environment. Symbol EMDEFS sets the EFS environment, and symbol NOEMDEFS resets the EFS environment.

C.4.2.3 TMP Directory

Enterprise Manager Agent run time creates several temporary files during processing. All temporary files are created in a directory specified by the logical name ORA_AGENT_TMP. This logical name is automatically defined to the physical path, ORA_ROOT:[SYSMAN.host.TMP], when setting up the Oracle environment.

It is possible that certain temporary files may not be deleted due to interruptions or failures. It is recommended that this directory content be monitored on a periodic basis and files that are more than three hours old be deleted.

C.4.2.4 Monitoring the Batch Queue

Section C.2, "Installation Requirements" mentions that the Enterprise Manager Agent runs as a batch job in a batch queue. Enterprise Manager Agent also submits several jobs to the batch queue for tasks such as monitoring, start and shut down of databases and listeners running in other Oracle installations. There is a possibility that broken environments in these installations could cause these submitted jobs to stop responding. It is recommended that the batch queue be periodically monitored for any long-pending jobs (any job of more than three hours is long-pending) and such jobs be deleted.

C.4.2.5 Disk Space on Enterprise Manager Agent Install Area

There are two agent parameters listed in the EMD.PROPERTIES properties file related to disk space:

- UploadMaxDiskUsedPct
- UploadMaxDiskUsedPctFloor

Ensure that the agent parameters are set correctly at the required level. When the percentage of space used on the disk exceeds UploadMaxDiskUsedPct, uploads will stop, and updates from the Enterprise Manager Agent to the Oracle Management Server. When uploads are stopped, Enterprise Manager Agent status will appear as up and running, but the line reporting Last Successful Upload will show a time stamp that remains the same over periods exceeding the value of UploadInterval in the EMD.PROPERTIES file. Uploads will be enabled again when the percentage of space used on the disk falls under the value of UploadMaxDiskUsedPctFloor, which should be less than UploadMaxDiskUsedPct.

C.4.2.6 Resetting Enterprise Manager Agent Environment

You can shut down Enterprise Manager Agent that is already running, and reconfigure it to point to a different Oracle Management Server. To reset Enterprise Manager Agent environment, it is recommended to perform the following tasks:

1. Delete the ORA_ROOT:[*host_sid*.SYSMAN.EMD]LASTUPLD.XML;* file.
2. Delete all files in the ORA_ROOT:[*host_id*.SYSMAN.EMD.UPLOAD] directory.
3. Delete all files in the ORA_ROOT:[*host_id*.SYSMAN.EMD.STATE] directory.
4. Delete all files in the ORA_ROOT:[*host_id*.SYSMAN.EMD.COLLECTION] directory.
5. Delete all files in the ORA_ROOT:[*host_sid*.SYSMAN.EMD.RECV] directory.
6. Rename or delete Enterprise Manager Agent log and trace files:

```
ORA_ROOT:[host_sid.SYSMAN.LOG]Enterprise Manager Agent.LOG
ORA_ROOT:[host_sid.SYSMAN.LOG]Enterprise Manager Agent.TRC
```

C.5 Supported Targets and Jobs

The following targets and jobs are supported by Enterprise Manager Agent:

Targets

Enterprise Manager Agent supports the following targets:

- Agent
- Host
- Database
- Listener
- Cluster Database (Oracle RAC)
- Cluster

When Enterprise Manager Agent is started, Enterprise Manager Agent reads the `TARGETS.XML` file in the `ORA_EMSTATE_EMD` directory, and registers those targets with the Oracle Management Server. The status of each of the targets is reflected under the Targets tab of the Enterprise Manager Console. Enterprise Manager Agent monitors the registered targets on a periodic basis and uploads the status of the targets to the Oracle Management Server, which is reflected on the Enterprise Manager Console. In addition, a set of predefined metrics are also collected for each target and uploaded to the Oracle Management Server. Default thresholds are defined for each predefined metric on the Oracle Management Server. When a threshold is reached, an alert is generated and displayed on the console. This mechanism of automatic target monitoring replaces the Events model of earlier releases.

> **See Also:** For more information about targets, see *Oracle Enterprise Manager Concepts*

Jobs

The following jobs are supported by Enterprise Manager Agent:

- Operating System Command
- SQL Script

C.6 Known Limitations

Enterprise Manager Agent has the following known limitations:

1. Jobs will always indicate a successful completion status, if the agent has been able to create a detached process and run the command specified for the job. The status of job completion is not a reflection of the completion status of the job command. If the command fails for some reason, then the job itself will not be indicated as a failure, but the output of the job must be checked to verify if the command has succeeded or not.
2. The metric browser does not support all metrics. All metrics in the host metrics area are not applicable to Oracle Database. For more details, refer to the `README_VMS_11GR2.TXT` file shipped with the Oracle Grid Control Enterprise Manager Agent Kit.

D

Managing Oracle Database Port Numbers

During installation, Oracle Universal Installer (OUI) 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:

D.1 About Managing Ports

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

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

- Is the port used by another Oracle Database installation on the host?

  The installation may be up or down at the time; Oracle Database can still detect if the port is used.

- Is the port used by a process that is currently running?

  This could be any process on the host, even a non-Oracle Database process.

- Is the port listed in the `TCPIP$ETC:SERVICES.DAT`?

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.

D.2 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 ORA_ROOT:[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.

D.3 Port Numbers and Protocols of Oracle Components

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

Table D–1 Ports Used in Oracle Components

Component and Description	Default Port Number	Port Range	Protocol
Oracle SQL*Net Listener Allows Oracle client connections to the database over Oracle's SQL*Net protocol. You can configure it during installation. To reconfigure this port, use Net Configuration Assistant.	1521	1521	TCP
Data Guard Shares the SQL*Net port and is configured during installation. To reconfigure this port, use Net Configuration Assistant to reconfigure the Oracle SQL*Net listener.	1521 (same value as the listener)	1521	TCP
Oracle Management Agent HTTP port for Enterprise Management Agent. It is configured during installation. "Changing the Oracle Enterprise Management Agent Port" on page D-2 explains how to modify its port number.	1832	1830–1849	HTTP

D.4 Changing the Oracle Enterprise Management Agent Port

To find the current setting for the Oracle Management agent port, search for EMD_URL in the ORA_ROOT:[*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
```

E

Configuring Oracle Database Globalization Support

This appendix describes these Globalization Support topics:

E.1 About NLS_LANG Logical

Oracle provides Globalization Support that enables users to interact with a database in their preferred locale and character set settings. Setting the NLS_LANG logical name specifies locale behavior for Oracle software. It sets the language and territory used by the client application. It also sets the character set for entering and displaying data by a client program, such as SQL*Plus.

The NLS_LANG logical uses the following format:

```
$ DEFINE NLS_LANG="language_territory.characterset"
```

where:

Parameter	Description
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 (normally the Oracle character set that corresponds to the character set of the user terminal or the operating system).

The choices of locales available on your system depend on the locales installed. For more information on HP OpenVMS locale refer to the Compaq C Run-Time Library Utilities Reference Manual.

The setting of NLS_LANG depends on the locale available in your session. Great care must be taken to set NLS_LANG correctly. An Incorrect NLS_LANG setting may lead to the input of incorrect data.

> **See Also:** *Oracle Database Globalization Support Guide* for information about the NLS_LANG parameter and Globalization Support initialization parameters

The following table lists some valid values for the NLS_LANG environment variable:

See Also: See the operating system specific documentation on how to determine the operating system locale environment setting

Operating system locale	`NLS_LANG` values
French (France)	FRENCH_FRANCE.WE8ISO8859P15, FRENCH_ FRANCE.WE8ISO8859P1, FRENCH_FRANCE.WE8MSWIN1252, FRENCH_FRANCE.AL32UTF8
Japanese (Japan)	JAPANESE_JAPAN.JA16EUC, JAPANESE_JAPAN.JA16SJIS, JAPANESE_JAPAN.AL32UTF8

F

Troubleshooting

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

- Section F.1, "Verifying Requirements"
- Section F.2, "X Window Display Errors"
- Section F.3, "Errors During Installation"
- Section F.4, "Reviewing the Log of an Installation Session"
- Section F.5, "Troubleshooting Configuration Assistants"
- Section F.6, "Silent Response File Error Handling"
- Section F.7, "Cleaning Up After a Failed Installation"

F.1 Verifying Requirements

Ensure that the system meets the requirements and that you have completed all of the preinstallation tasks specified in Chapter 2, "Preinstallation Tasks" before you perform any of the troubleshooting steps in this appendix.

Read the Release Notes

`http://docs.oracle.com/en/database/database.html`

F.2 X Window Display Errors

If you run Oracle Universal Installer on a remote system and you want to display the Oracle Universal Installer user interface on a local system, you may see error messages similar to the following:

```
%DECW-E-CANT_OPEN_DISPL, Can't open display
```

If you see the preceding error message, and the X server is running on a UNIX host, then:

1. In a local terminal window, log in as the user that started the X Window session.
2. Enter the following command:

   ```
   $ xhost +
   ```

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 or tcsh 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 be displayed on the monitor.

5. If the X clock is displayed, then close X clock and start Oracle Universal Installer again.

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.

F.3 Errors During Installation

If you encounter an error during installation:

- Do not exit Oracle Universal Installer.
- If you clicked **Next** after you entered incorrect information about one of the installation screens, click **Back** to return to the screen and correct the information.
- See Section F.4, "Reviewing the Log of an Installation Session" if you encounter an error while Oracle Universal Installer is copying or linking files.
- See Section F.5, "Troubleshooting Configuration Assistants" if you encounter an error while a configuration assistant is running.
- Remove the failed installation by following the steps listed in Section F.7, "Cleaning Up After a Failed Installation" if you cannot resolve the problem.

F.4 Reviewing the Log of an Installation Session

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

To view the log file:

1. Determine the location of the `oraInventory` directory. This can be found as the value of the `inventory_loc` parameter in the `ORAINST.LOC` file, which is located in the *hostname* subdirectory of `SYS$LOGIN`.
2. Set default to the directory determined in step 1, and then to the `LOGS` subdirectory located there.
3. Obtain a directory listing to determine the file name of the log file. Installer log files have names similar to the following, where `date_time` indicates the date and time that the installation started:

```
INSTALLACTIONSdate_time.LOG
```

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. If an error displayed by Oracle Universal Installer or listed in the log file indicates a relinking problem, refer to the following file for more information:

```
ORA_ROOT:[INSTALL]MAKE.LOG
```

F.5 Troubleshooting Configuration Assistants

To troubleshoot an installation error that occurs when a configuration assistant is running:

- See Section F.4, "Reviewing the Log of an Installation Session" lists the installation log files for review.
- Review the specific configuration assistant log file located in the `ORA_ROOT:[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. See Section F.5.2, "Fatal Errors" for more information.

F.5.1 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:

```
device:[orainventory_location.LOGS]SILENTINSTALLdate_time.LOG
```

Table F–1 lists the execution status codes:

Table F–1 Execution Status Codes

Status	Result Code
Configuration assistant succeeded	0
Configuration assistant failed	1
Configuration assistant canceled	-1

F.5.2 Fatal Errors

If you receive a fatal error while a configuration assistant is running, you must remove the current installation and reinstall the Oracle software as follows:

1. See Section F.7, "Cleaning Up After a Failed Installation" to remove failed installations.
2. Correct the cause of the fatal error.
3. Reinstall the Oracle software.

F.6 Silent Response File Error Handling

To determine whether a silent installation succeeds or fails, refer to the following log file:

```
device:[orainventory_location.LOGS]SILENTINSTALLdate_time.LOG
```

If necessary, see Section F.4, "Reviewing the Log of an Installation Session" for information about determining the location of the `oraInventory` directory.

A silent installation fails if:

- You do not specify a response file
- You specify an incorrect or incomplete response file
- Oracle Universal Installer encounters an error, such as insufficient disk space

Oracle Universal Installer or configuration assistant validates the response file at run time. If the validation fails, then the silent 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.

F.7 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. See Section 3.3.1, "Installing Oracle Database from the Hard Drive" to start Oracle Universal Installer.
2. Click **Deinstall Products** on the Welcome screen or click **Installed Products** on any Installer screen.

   The Inventory screen is displayed, 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.

G

Apache Server Configuration

This appendix lists the procedure and steps to configure Apache Server for HP OpenVMS. The following topics are included:

- Section G.1, "Postinstallation Checklist"
- Section G.2, "Test the Installation"
- Section G.3, "Running Oracle Apache Server on HP OpenVMS"
- Section G.4, "Security Information"
- Section G.5, "Open Source Licenses"

G.1 Postinstallation Checklist

After you configure Oracle HTTP Server for HP OpenVMS, perform the following tasks to ensure a successful startup:

- Section G.1.1, "Running AUTOGEN"
- Section G.1.2, "Check Disk Quota"
- Section G.1.3, "Checking for SET TERMINAL/INQUIRE"

Each of these tasks is explained in the following sections. Once you have completed these, you can test the installation by starting Oracle HTTP Server.

G.1.1 Running AUTOGEN

After the installation, when you have a normal system workload running on your system, run `SYS$UPDATE:AUTOGEN.COM (AUTOGEN)` to evaluate the system parameters and make adjustments based on the hardware configuration and system workload. On Oracle HTTP Server, `AUTOGEN` will probably increase the page file size and the number of swap file pages.

G.1.2 Check Disk Quota

If the disk quota is too low, then Oracle HTTP Server will not start. Either raise the disk quota for the Oracle Database account or grant the account the `EXQUOTA` privilege, which enables it to bypass disk quota restrictions. Use the following commands:

```
$ SHOW QUOTA/USER=[server-uic]/DISK=device_name
$ SET PROCESS/PRIVILEGES=EXQUOTA node-name::ORACLE
```

G.1.3 Checking for SET TERMINAL/INQUIRE

When Oracle HTTP Server for HP OpenVMS is started, the following login files are run:

- SYS$MANAGER:SYLOGIN.COM
- LOGIN.COM (login for the Oracle Database account)

Check these files to ensure that any SET TERMINAL/INQUIRE statements are processed only in the INTERACTIVE mode. For example:

```
$ IF F$MODE() .EQS "INTERACTIVE" THEN $ SET TERMINAL/INQUIRE
```

If you do not check this, then the HTML that is sent to clients may not be well-formed and may be sent in an intermittent fashion. This problem may also appear when running CGI scripts.

G.2 Test the Installation

You must manually start Oracle HTTP Server to verify the installation and configuration of the server. Enter the following command:

```
$ APACHECTL START
```

Perform the following tasks to test the installation:

- Section G.2.1, "Browser Test"
- Section G.2.2, "Troubleshooting"

G.2.1 Browser Test

You can test the installation using a Web browser. Replace *host.domain* in the following URL with the corresponding information about Oracle HTTP Server that you installed:

```
HTTP://host.domain:port
```

If this is a new installation, then the browser should display the standard introductory screen with the following bold text at the top:

```
"Welcome to Oracle HTTP Server."
```

The Apache logo is displayed at the bottom.

G.2.2 Troubleshooting

If you do not receive a response from Oracle HTTP Server, check the following:

- In the SYS$MANAGER:SYLOGIN.COM file, ensure that there is no SET TERMINAL/INQUIRE statement for network processes.
- Look for the following files:

  ```
  APACHE$ROOT:[000000]APACHE$SERVER.LOG
  APACHE$ROOT:[LOGS]ERROR_LOG
  ```

G.3 Running Oracle Apache Server on HP OpenVMS

The default port for Oracle HTTP Server is port 7777. If this port is already in use by another application or to use a different port, then modify the HTPPD.CONF file,

located in the ORA_ROOT:[APACHE.APACHE.SPECIFIC.host.CONF] directory, to specify a different port number.

The following subsections describes the process of running the Oracle HTTP server:

G.3.1 Starting and Stopping the Server

To start Oracle HTTP Server, enter the following command:

```
$ APACHECTL STARTUP
```

To stop Oracle HTTP Server, enter the following command:

```
$ APACHECTL STOP
```

G.3.2 Server Log File

The server log file for APACHE$WWW is written as:

```
APACHE$SPECIFIC:[000000]APACHE$SERVER.LOG
```

G.3.2.1 Performance Considerations

You should have prior experience in tuning the performance of the HP OpenVMS operating system. For information about HP OpenVMS performance, see HP OpenVMS system documentation.

Recommendations for improving performance on Oracle HTTP Server are provided in this appendix and *Oracle Database Release Notes for HP OpenVMS Itanium*.

G.3.2.2 Limits and Quotas for Light to Moderate Load

Table G–1 shows sample values for the Oracle Database account from a working and exercised Oracle HTTP Server with a light to moderate load. These values are presented as an example of a system performing well within its context.

If you should experience performance difficulties, refer to this table for guidelines about making adjustments. Remember that no one set of values will be appropriate for all situations.

Table G–1 Sample Values for the oracle account

Parameter	Default	Sample value for Oracle HTTP Server
ASTLM (NonPooled) Total number of asynchronous system trap (AST) operations and scheduled wake-up requests the user can have queued simultaneously.	250	610 Or BIOLM + DIOLM + 10
BIOLM (NonPooled) Number of outstanding buffered I/O operations permitted for a user process.	150	300 You may also need to increase the SYSGEN parameter CHANNELCNT because it limits BIOLM,DIOLM, and FILLM.
BYTLM (Pooled) Amount of buffer space a user process can use.	64000	200000 Increase this value for a heavy load.
CHANNELCNT	256	256 CHANNELCNT must be greater than or equal to FILLM
DIOLM (NonPooled) Number of outstanding direct I/O operations permitted for a user process.	150	300 You may also need to increase the SYSGEN parameter CHANNELCNT because it limits BIOLM,DIOLM, and FILLM.
ENQLM (Pooled) Specifies the lock queue limit.	2000	2000
FILLM (Pooled) Number of files a user process can have open simultaneously. This includes the number of network logical links that can be active at the same time.	100	300 Increase this value for a heavy load. You may also need to increase the SYSGEN parameter CHANNELCNT because it limits BIOLM,DIOLM, and FILLM.
JTQUOTA (Pooled) Byte quota for the job-wide logical name table.	4096	8192
PGFLQUO (Pooled) Number of pages the user process can use in the system page file.	50000	250000 If you increase PGFLQUO, then you should monitor the free size of the system page and swap files, because these may need to be increased.
PRCLM (Pooled) Number of subprocesses a user process can create.	8	20 You should increase this value for a heavy load.
TQELM (Pooled) Number of entries a user process can have in the timer queue or the number of temporary common event flag clusters a user process can have.	10	610 Or BIOLM + DIOLM + 10

G.3.2.3 Server with Medium to High Load

After you install the server and run it, look in the log file for errors of the "cannot open" type. Errors of this type often indicate that you must modify system parameters. Try the following:

- Set FILLM to limit the number of files that a user process can have open.
- Set the SYSGEN parameter, CHANNELCNT, to 1024 (unless it is already set to a higher value).

> **Note:** Whenever you change system parameters, you must restart the system to enable the new settings.

G.3.2.4 Excessive File Build Up

A large number of .LOG and .PID files can amass over time in the APACHE$ROOT:[0000000] and APACHE$ROOT:[LOGS] directories. Purging these files can become a burden on application or system managers. System managers should manually use explicit SET DIRECTORY/VERSION commands on these two directories.

G.3.3 Customizing the Server Environment

The installation procedure creates a file named HTTPD.CONF and places it in APACHE$ROOT:[CONF]. The HTTPD.CONF file stores information that Oracle HTTP Server uses to set up the server environment. The HTTPD.CONF file has been tailored to use HP OpenVMS syntax, but its overall functionality is essentially identical to httpd.conf on the UNIX platform.

The HTTPD.CONF file contains an explanation for each line that it can process. You can refer to these explanations when customizing the file for your environment. You can also refer to any generally available Apache documentation on HTTPD.CONF.

Note the following about HTTPD.CONF on HP OpenVMS:

- MOD_OSUSCRIPT has been added to enable CGI scripts that were originally written for the OSU server.
- UNIX-style path names are recognized by HP OpenVMS. You can use either UNIX-style or HP OpenVMS-style path names in the configuration file. However, you cannot mix the two styles within a specification.
- In an HP OpenVMS Cluster, you can specify either clusterwide or system-specific files.

G.3.4 How to Configure Apache

Perform the following steps to configure Apache to run while using an account other than the Oracle Database account:

1. Modify the HTTPD.CONF file to include the following line:

   ```
   User username
   ```

2. Modify APACHEUSER.COM to set the logical APACHE_USERNAME to the required user name. Ensure that the user name is the same as the entry in the HTTPD.CONF file.
3. Restart the Apache Server if it is already running.

G.3.5 Modules and Directives

The Oracle HTTP Server provides the modules and directives that are provided by the HP Secure Web Server for HP OpenVMS (based on Apache), Version 1.3-1. See the HP documentation for that product for more information. All supported modules and directives function as documented by the Apache Software Foundation at

http://httpd.apache.org/docs/

G.3.6 Supported and Unsupported Features

Information about running Oracle HTTP Server that is specific to running the server on HP OpenVMS is provided in the following sections.

> **See Also:** For more information about Apache server, refer to the Apache Software Foundation website at
>
> http://httpd.apache.org/docs/

G.3.6.1 Modules Not Included

The following modules are not included in this version of Oracle HTTP Server:

- MOD_OSNINT
- MOD_OSSLl
- MOD_PERL
- MOD_PHP
- MOD_PROXY

G.3.6.2 Unsupported Directives

The following directives are not supported:

- AgentLog
- AllowCONNECT
- Anonymous
- Anonymous_Authoritative
- Anonymous_LogEmail
- Anonymous_MustGiveEmail
- Anonymous_NoUserID
- Anonymous_VerifyEmail
- AuthDBAuthoritative
- AuthDBGroupFile
- AuthDBMAuthoritative
- AuthDBMGroupFile
- AuthDBUserFile
- AuthDBMUserFile
- AuthDigestFile
- CacheDefaultExpire

- CacheDirLength
- CachedirLevels
- CacheForceCompletion
- CacheGcInterval
- CacheLastModifiedFactor
- CacheMaxExpire
- CacheRoot
- CacheSize
- CheckSpelling
- CookieExpires
- CookieTracking
- Example
- ExpiresActive
- ExpiresByType
- ExpiresDefault
- Header
- Metadir
- MetaFiles
- MetaSuffix
- MimeMagicFile
- MMapFile
- NoCache
- ProxyBlock
- ProxyDomain
- ProxyPass
- ProxyPassReverse
- ProxyReceiveBufferSize
- ProxyRemote
- ProxyRequests
- ProxyVia
- RefererIgnore
- RefererLog
- RewriteBase
- RewriteCond
- RewriteEngine
- RewriteLock
- RewriteLog

- RewriteLogLevel
- RewriteMap
- RewriteOptions
- RewriteRule
- ScriptInterpreterSource
- VirtualDocumentRoot
- VirtualDocumentRootIP
- VirtualScriptAlias
- VirtualScriptAliasIP

> **See Also:** For information about these directives, see the HP OpenVMS documentation for HP Secure Web Server (based on Apache, version 1.3-1)

G.3.6.3 Command-Line Options

This section describes the HTTPD command-line options supported on Oracle HTTP Server.

Then you can use the following format to enter a command-line option:

```
$ HTTPD -option
```

where -option is one of the following command line options:

- "-v"

  Displays the HTTPD version and its build date.

- "-V"

  Displays the HTTPD base version, its build date, and a list of compile settings that influence the performance of the server.

- -h:

  Displays a list of the HTTPD options.

- "-l":

  Displays a list of all modules compiled into the server.

- -"L":

  Displays a list of directives with expected arguments and instances where the directive is valid.

The following example shows how to enter the "L" option to list the available configuration directives:

```
$ HTTPD "-L"
```

G.3.6.4 Virtual Host Support

The term, **virtual host**, refers to the practice of maintaining a single server to serve pages for multiple virtual hosts. Both IP-based and name-based virtual host support are available on Oracle HTTP Server for HP OpenVMS.

Note: The security profile of the running server is the same on all virtual hosts.

For more information about virtual hosts, see the Apache Software Foundation documentation at

http://httpd.apache.org/docs/current/vhosts/index.html

G.3.6.5 Dynamic Shared Object Support

Dynamic shared object support provides a method to format code so that it will load into the address space of an executable program at run time. For more information about dynamic shared object support, see the Apache Software Foundation documentation at

http://httpd.apache.org/docs/current/dso.html

G.3.6.6 File Handlers

Oracle HTTP Server for HP OpenVMS supports the ability to use file handlers explicitly. For more information about file handlers, see the Apache Software Foundation documentation at

http://httpd.apache.org/docs/current/handler.html

G.3.6.7 Content Negotiation

The `MOD_NEGOTIATION` module provides content negotiation. This module enables you to specify language variants of HTML files. To specify language variants, use an underscore instead of a period before the language extension.

For example:

- On UNIX, `filename.html.fr` is the French variant of `filename.html`.
- On HP OpenVMS, `FILENAME.HTML_FR` is the French variant of `FILENAME.HTML`.

For more information about content negotiation, see the Apache Software Foundation documentation at

http://httpd.apache.org/docs/current/content-negotiation.html

G.3.6.8 Apache API

You can use the standard Apache application programming interface (API) to write user-defined modules that run on Oracle HTTP Server. For more information about the Apache API, see the Apache Software Foundation documentation at

http://httpd.apache.org/docs/

G.3.6.9 suEXEC Support

The suEXEC feature provides the ability to run CGI programs under user IDs that are different from the user ID of the calling Web server. This is not supported by Oracle HTTP Server for HP OpenVMS.

G.3.7 File Formats

All file formats are supported. However, the Web browser status bar will not show page loading progress for Variable or VFC format files larger than 8 KB.

Page loading progress relies on an accurate byte count, which is not readily available for files in Variable or VFC format. For files in these formats, Oracle HTTP Server must count the bytes as the files load. The counting process can slow performance, so it has been turned off in this situation.

G.3.8 File Naming Conventions

In general, users running Oracle HTTP Server for HP OpenVMS can specify either UNIX-style file names or HP OpenVMS-style file names. Oracle HTTP Server usually displays UNIX-style file names.

The On-Disk Structure Level 5 (ODS-5) volume structure, shipped with HP OpenVMS version 8.2, supports long file names, enables the use of a wider range of characters within file names, and preserves case within file names. However, the DEC C RTL that is shipped with version 7.2-1 does not provide full support for extended file names on ODS-5 devices. This lack of full support imposes certain restrictions on users running Oracle HTTP Server for HP OpenVMS.

Because mixed UNIX-style and HP OpenVMS-style extended file names are not yet supported by the DEC C RTL, you may be required to use UNIX-style syntax when interacting with Oracle HTTP Server. An example would be appending additional directories or a file name to a root.

The following examples illustrate mixed UNIX-style and HP OpenVMS-style file names that are not supported in HP OpenVMS version 8.2:

```
doc/foo.bar.bar
./tmp/foo.bar.b^_ar
~foo^.bar
```

You can, however, modify the last example so that it works as an HP OpenVMS extended file name that has a tilde (~) as the first character. Precede the leading tilde (~) with the Extended File Specifications escape character (^) as shown in the following example:

```
^~foo^.bar
```

For more information about using the tilde (~) in HP OpenVMS extended file names, see the HP OpenVMS Guide to Extended File Specifications at

http://h71000.www7.hp.com/doc/73final/6536/6536pro.html

G.3.9 File Transfer Process and Access Control List

When performing an FTP operation, ensure that the access control list (ACL) for the target directory on Oracle HTTP Server has FTP access enabled as follows:

When transferring new files:

```
$ SET SECURITY/ACL=(IDENTIFIER=yourFTPname,ACCESS=READ+WRITE) [directory]
```

When replacing existing files:

```
$ SET SECURITY/ACL=(IDENTIFIER=yourFTPname,ACCESS=READ+WRITE) [directory]*.*
```

G.3.10 Logical Names

Table G–2 lists the following logical names created by Oracle HTTP Server for HP OpenVMS:

Table G–2 Oracle HTTP Server Logical Names and Their Descriptions

Logical Name	Description
APACHE$COMMON	Concealed logical name that defines clusterwide files in APACHE$ROOT (device:[APACHE]).
APACHE$FIXBG	System executive mode logical name pointing to installed, sharable images. Not intended to be modified by the user.
APACHE$HTTPD_SHR	System executive mode logical name pointing to installed, sharable images. Not intended to be modified by the user.
APACHE$INPUT	Used by CGI programs for PUT and POST methods of reading the input stream.
APACHE$PLV_ENABLE_*username*	System executive mode logical name defined during startup and used to control access to the services provided by the APACHE$PRIVILEGED image. Not intended to be modified by the user.
APACHE$PLV_LOGICAL	System executive mode logical name defined during startup and used to control access to the services provided by the image. Not intended to be modified by the user.
APACHE$PRIVILEGED	System executive mode logical name pointing to installed, sharable images. Not intended to be modified by the user.
APACHE$ROOT	System executive mode logical name defined during startup that points to the top-level directory. (device:[APACHE], device:[APACHE.SPECIFIC.*node-name*]).
APACHE$SPECIFIC	Concealed logical name that defines system-specific files in APACHE$ROOT (device:[APACHE.SPECIFIC.*node-name*]).
APACHE$CGI_MODE	System logical name that controls how CGI environment logicals are defined in the running CGI process. There are three different options. Only one option is available at a time. 0: Default. Environment logicals are defined as local symbols and are truncated at 970 (limitable with DEC C). 1: Environment logicals are defined as local symbols unless they are greater than 970 characters. If the environment value is greater than 970 characters, then it is defined as a multi-item logical. 2: Environment logicals are defined as logicals. If the environment value is greater than 512 characters, then it is defined as a multi-item logical.
APACHE$DEBUG_DCL_CGI	If defined, this system logical name enables APACHE$VERIFY_ DCL_CGI and APACHE$SHOW_CGI_ SYMBOL.
APACHE$VERIFY_DCL_CGI	If defined, this system logical name provides information for troubleshooting DCL command procedure CGIs by forcing a SET VERIFY before running any DCL CGI. Use with APACHE$DEBUG_DCL_CGI.
APACHE$SHOW_CGI_SYMBOL	If defined, this system logical name provides information for troubleshooting the CGI environment by dumping all the symbols and logicals (job/process) for a given CGI. Use with APACHE$DEBUG_DCL_CGI.

Table G–2 (Cont.) Oracle HTTP Server Logical Names and Their Descriptions

Logical Name	Description
APACHE$PREFIX_DCL_CGI_SYMBOLS_WWW	If defined, this system logical name prefixes all CGI environment logical symbols with WWW_. By default no prefix is used.
APACHE$CREATE_SYMBOLS_GLOBAL	If defined, this system logical name causes CGI environment symbols to be defined globally. They are defined locally by default.
APACHE$CGI_USE_DCLCOM_FOR_IMAGES	If defined, this system logical name forces CGI images to run within a DCL process. The default is to run CGI images directly. (**Note:** Direct running of CGI images in not currently supported).
APACHE$DL_NO_UPPERCASE_FALLBACK	If defined to be true (1, T, or Y), this system logical name disables case-insensitive symbol name lookups whenever case-sensitive lookups fail. Refer to APACHE$DL_FORCE_UPPERCASE.
APACHE$DL_FORCE_UPPERCASE	If defined to be true (1, T, or Y), this system logical name forces case-sensitive dynamic image activation symbol lookups. By default, symbol lookups are first done in a case-sensitive manner, and then if failed, a second attempt is made by using case-insensitive symbol lookups. This fallback action can be disabled with APACHE$DL_NO_UPPERCASE_FALLBACK.

G.3.11 HP OpenVMS Cluster Considerations

An HP OpenVMS Cluster is a group of HP OpenVMS systems that work together as one virtual system. Oracle HTTP Server runs in an HP OpenVMS Cluster so that you can take advantage of the resource sharing that increases the availability of services and data.

Bear the following points in mind:

- Oracle HTTP Server is supported on HP OpenVMS Version 8.2-1 or later.
- Oracle HTTP Server runs in an Itanium, or in a mixed architecture cluster, separate Apache installations are required for Itanium.

G.3.11.1 Individual System Versus Clusterwide Definition

To define clusterwide versus individual configuration files, APACHE$ROOT uses the following concealed logical names:

- APACHE$COMMON defines clusterwide files.
- APACHE$SPECIFIC defines system-specific files.

When reading a file, the server first looks for a system-specific version of the file in APACHE$SPECIFIC:[*directory*]. If it does not find one, then it looks for a clusterwide file in APACHE$COMMON:[*directory*].

To avoid confusion, always use the appropriate concealed logical name to specify the file that you want to edit. For example, to edit a clusterwide version of HTTPD.CONF, see:

```
$ EDIT APACHE$COMMON:[CONF]HTTPD.CONF
```

If you refer to:

```
$ EDIT APACHE$ROOT:[CONF]HTTPD.CONF
```

then the server would open the clusterwide file but save it as a system-specific version. The latest version of HTTPD.CONF would then be visible only to the individual node on which it was saved.

Within HTTPD.CONF itself, you should make this distinction whenever you refer to a path or to a file location. This improves performance and ensures that the server will return a complete directory listing. For example, you should specify APACHE$COMMON or APACHE$SPECIFIC (instead of APACHE$ROOT) with directory directives.

The following extract, from the HTTPD.CONF file, refers to APACHE$COMMON, because the content for the default Web page is in the clusterwide directories.

```
DocumentRoot "/apache$common/htdocs"
        ...
      <Directory "/apache$common/htdocs">
      Options Indexes FollowSymLinks Multiviews
      AllowOverride None
      Order allow,deny
      Allow from all
      </Directory>
```

If there was content for one specific node in a cluster, then the APACHE$SPECIFIC logical name would be used.

G.3.11.2 Mixed-Architecture Cluster

In a mixed-architecture cluster containing VAX nodes, do not use a cluster alias IP address with Oracle HTTP Server. Because the VAX systems will not have Oracle HTTP Server running, they will not be able to service HTTP requests.

G.3.12 CGI Programs

Common gateway interface (CGI) programs run within the DCL shell on Oracle HTTP Server for HP OpenVMS. This section discusses the following CGI topics:

- Section G.3.12.1, "CGI Environment Logical"
- Section G.3.12.2, "Referencing Input"
- Section G.3.12.3, "Running CGI Images"
- Section G.3.12.4, "Logical Names for Debugging CGI Scripts"
- Section G.3.12.5, "Displaying Graphics with CGI Command Procedures"

G.3.12.1 CGI Environment Logical

By default, an environment logical symbol takes the form that is designated by the environment logical. You can determine how environment logicals are set when the server runs a CGI program. You can define the APACHE$PREFIX_DCL_CGI_SYMOBLS_WWW logical name to prefix all environment logical symbols with WWW_. By default, no prefix is used.

The APACHE$CGI_MODE logical name controls how CGI environment logicals are defined in the running CGI program as follows:

```
APACHE$CGI_MODE      option
```

where option can have one of the following values at a time:

- 0: Default. Environment logicals are defined as local symbols and are truncated at 970 (limitable with DEC C).

- 1: Environment logicals are defined as local symbols unless they are greater than 970 characters. If the environment value is greater than 970 characters, it is defined as a multi-item logical.
- 2: Environment logicals are defined as logicals. If the environment value is greater than 512 characters, it is defined as a multi-item logical.

`APACHE$DCL_ENV` is a foreign symbol that lets you define CGI environment logical, as follows:

```
APACHE$DCL_ENV [-c] [-d] [-e env-file]
```

where:

- `-c`: Default. Indicates create environment logicals.
- `-d`: Indicates delete environment logicals.
- `-e env-file`: Specifies an alternate environment file.

  The environment file does not need to be specified by the caller because the parent derives it (it can be easily determined by default).

The following example deletes the environment and then re-creates it:

```
Example: diff_mode_cgi.com
$ APACHE$DCL_ENV -d
$ Define APACHE$PREFIX_DCL_CGI_SYMBOLS_WWW 1
$ APACHE$DCL_ENV -c
```

G.3.12.2 Referencing Input

CGI scripts that reference input to Oracle HTTP Server must refer to `APACHE$INPUT`.

G.3.12.3 Running CGI Images

On HP OpenVMS, CGI images run within a DCL process. You cannot run CGI images directly.

G.3.12.4 Logical Names for Debugging CGI Scripts

Use the following logical names to debug CGI scripts:

Logical Name	Description
APACHE$DEBUG_DCL_CGI	If defined, this system logical name enables APACHE$VERIFY_DCL_CGI and APACHE$SHOW_CGI_SYMBOL.
APACHE$VERIFY_DCL_CGI	If defined, this system logical name provides information for troubleshooting DCL command procedure CGIs by forcing a SET VERIFY before running any DCL CGI. Enabled by APACHE$DEBUG_DCL_CGI.
APACHE$SHOW_CGI_SYMBOL	If defined, this system logical name provides information for troubleshooting the CGI environment by dumping all of the symbols and logicals (job/process) for a given CGI. Enabled by APACHE$DEBUG_DCL_CGI.

G.3.12.5 Displaying Graphics with CGI Command Procedures

To display a graphics file with a CGI command procedure, use the `APACHE$DCL_BIN` foreign symbol in the following format:

```
APACHE$DCL_BIN [-s bin-size] bin-file
```

where:

- `-s bin-size`: Specifies the actual or approximate file size in bytes. The value of `bin-size` is automatically determined if the image file is larger than 32768 KB (default value). If the image file is smaller than 32768 KB, then you can provide an approximate (or actual) size. This boosts performance.
- `bin-file`: Specifies the file to be displayed.

For example:

```
$ SAY := WRITE SYS$OUTPUT
$ SAY "Content-type: image/gif"
$ SAY ""
$ APACHE$DCL_BIN APACHE$ROOT:[ICONS]APACHE_PB.GIF
$ EXIT
```

G.4 Security Information

Oracle HTTP Server for HP OpenVMS is a nonprivileged, user-mode, socket-based network application. `TMPMBX` and `NETMBX` are the only privilege requirements. The server runs under its own unique UIC and user account (`APACHE$WWW`).

G.4.1 Process Model

Oracle HTTP Server runs as a single job that consists of:

- One master process (`APACHE$WWW`).
- Several subprocesses.

  Subprocesses are created to service incoming HTTP requests and to run CGI scripts.

Because the server runs as a single job, the HP OpenVMS security profile for each process is identical and no enhanced mechanism is required for these processes to communicate with one another. A single user account (`oracle`) controls the resource utilization where pooled quotas are defined.

G.4.2 Privileged Images

Oracle HTTP Server performs three operations that require additional privileges:

- Binding to a port under 1024 (privileged ports).

  By default, the server binds to port 8080 (HTTP).
- Fetching path information for other users.

  The server provides a replacement for the `getpwnam` C RTL routine to enable the server to fetch default path information for other users (required by `MOD_UTIL` and `MOD_USERDIR`).
- Changing the carriage-control attribute on socket (BG) devices.

  The server also enables or disables (or both) the carriage-control attribute on BG (socket) devices for certain stream operations.

Two protected, sharable images are installed at startup to enable the server to perform the following functions:

- `APACHE$PRIVILEGED.EXE` (exec-mode services).

- APACHE$FIXBG.EXE (kernel-mode services).

The APACHE$PRIVILEGED.EXE image provides exec-mode services for binding to privileged sockets and fetching user default path information. Access to these services is limited to processes running under the oracle username. The APACHEPLV_ENABLE_APACHEWWW logical name controls these services. This logical name is defined as:

```
"APACHE$PLV_ENABLE_APACHE$WWW" = "3,80,1023"
```

The "3,80,1023" string represents three parameters where:

- The first parameter (3) is a bit-mask that enables or disables the two services:
  - Binding to privileged ports.
  - Fetching user default path information.
- The second and third parameters indicate the minimum and maximum port that are allowed to be bound.

When a call to either service is made, the service code does the following:

1. Temporarily enables the SYSPRV, OPER, SYSNAM, and NETMBX privileges.
2. Performs the function.
3. Restores the process original privileges.

The APACHE$FIXBG.EXE image provides a kernel-mode service for manipulating the carriage-control attribute for BG devices that are owned by the calling process. No special access control exists on this service. This function can also be performed using a setsocketopt C RTL run-time call, but it is not supported by all TCP/IP stack vendors, which is the reason this service exists. This service does not enable privileges, but runs in kernel mode.

G.4.3 Privileges Required to Start and Stop the Server

Oracle HTTP Server runs under the oracle username and UIC and is started as a detached, network process. During startup, protected images are installed and logical names are placed in the system logical name table. Shutdown is accomplished by sending a KILL signal to the master process and its subprocess.

In order to startup the Oracle HTTP Server, the following privileges are required:

- SYSPRV
- SYSNAM
- IMPERSONATE
- BYPASS
- CMKRNL
- ALTPRI
- WORLD

The privileges ALTPRI and BYPASS are not essential privileges for installing and running an Oracle database. These privileges can either be added to the oracle account, or a separate account can be created to maintain and run the Oracle HTTP Server.

If you have already configured Oracle HTTP Server using the oracle account and want to run under a different account, then the file ORA_

ROOT:[APACHE.APACHE.SPECIFIC.node.CONF]HTTPD.CONF must be modified to change the USER parameter before attempting to start it up.

G.4.4 File Ownership and Protection

All the server files reside under the root directories that the APACHE$ROOT logical name points to. During installation, file protection is set to (S:RWED, O:RWED, G, W). During configuration, all files are set to be owned by the oracle user.

G.4.5 Server Extensions (CGI Scripts)

Server extensions, such as CGI scripts, run within the context of Oracle HTTP Server process or its subprocesses. These extensions have complete control over the server environment. You can configure the server to enable processing of arbitrary user scripts, but standard practice is to limit such activity to scripts that are written by completely trusted users. Oracle HTTP Server includes directives that enable a Web administrator to control script execution and client access. The use of these directives are described in numerous books and is not duplicated here.

G.4.6 suEXEC Not Available for Protecting Script Execution

Oracle HTTP Server for HP OpenVMS does not currently support the suEXEC method of running scripts under the username that owns the script. Many sites use this feature to allow execution of arbitrary, user-written scripts without the fear of compromising the server environment.

G.5 Open Source Licenses

This section provides open source license acknowledgments and license references.

G.5.1 Apache

This product includes software developed by the Apache Software Foundation. You can visit the website of this organization at

http://www.apache.org/

You can view the license at

http://www.apache.org/licenses/LICENSE-2.0

This product also includes software that is developed by Hewlett-Packard.

H

Identifying Listener Processes

Use the following set of commands to display information about all listener processes:

```
$!
$!  To identify and report on any active SQL*Net listener processes
$!  on this OpenVMS Server (used as part of 11g pre-installation
$!  check activities)
$!
$ found_any = 0
$ say = "write sys$output"
$!
$ say " "
$ say " Collecting system information - please wait..."
$!
$!  First, get a list of all disk on this system
$!
$ pipe show dev d/mount > sys$scratch:lsnrchk_devices.tmp
$!
$ close/nolog flsnr
$ open/read flsnr sys$scratch:lsnrchk_devices.tmp
$!
$!  Skip the records we know we don't need
$!
$ read flsnr record
$ read flsnr record
$ read flsnr record
$!
$!  Loop on each disk we found
$!
$ loop:
$ read flsnr/end=done_it record
$ dev = f$element(0," ", record)
$!
$!  Look for any processes accessing tnslsnr.exe
$!
$ pipe show device/files 'dev' | search/nowarn -

$ sys$input tnslsnr > sys$scratch:lsnr_chk1.tmp
$!  If we found something, check the process(es) returned
$!
$ if $status .eqs. "%X10000001"
$ then
$    close/nolog tmmp
$    open/read tmmp sys$scratch:lsnr_chk1.tmp
$loop1:
$    read/end=done_it_1 tmmp record
$    if record .nes. ""
```

```
$    then
$       record = f$edit(record,"COMPRESS,TRIM,UPCASE")
$! Skip tnslsnr.com records
$       if f$locate(".COM;",record) .ne. f$length(record) then goto loop1
$       proc = f$element(1," ",record)
$!
$!  The following check ensures we skip records that we don't want
$!  (for example, records that reflect that we have a installed image)
$!
$       if proc .nes. "" .and. f$locate(".EXE",proc) .eq. f$length(proc)
$       then
$          img = f$getjpi("''proc'","IMAGNAME")
$          lsnr_loc = f$extract(0,f$locate(".]",img),img)
$          lsnr_loc = f$extract(0,f$locate(".NETCONFIG]",lsnr_loc),lsnr_loc)
$          lsnr_loc = lsnr_loc  + ".NETWORK.ADMIN]"
$          prcnm = f$getjpi("''proc'","PRCNAM")
$!
$!  Display the result
$!
$          if found_any .eq. 0
$          then
$             say " "
$             say " The following listener(s) need to be checked :"
$             say " "
$             say " If they use the IPC Protocol with (key = EXTPROC) and/or "
$             say " the TCP Protocol with (Port = 1521) then they need to "
$             say " be shutdown when installing Oracle Database 11g"
$             say " "
$          endif
$          say "=============="
$          say "  Listener process name          : ''prcnm'"
$          say "  VMS pid                        : ''proc'
$          say "  Probable LISTENER.ORA location : ''lsnr_loc'"
$          found_any = found_any + 1
$       endif
$       goto loop1
$    endif
$ done_it_1:
$    close/nolog sys$scratch:lsnr_chk1.tmp
$ endif
$ goto loop
$!
$ done_it:
$ close/nolog flsnr
$!  Remove all work files
$!
$ delete/noconfirm sys$scratch:lsnrchk_devices.tmp;*
$ delete/noconfirm sys$scratch:lsnr_chk1.tmp;*
$!
$ if found_any .eq. 0
$ then
$    say " "
$    say "       No active SQL*Net Listeners detected on this node"
$    say " "
$ else
$    say "=============="
$    say " "
$ endif
$!
$ say " End of processing"
```

```
$ say " "
$ exit
```

Index

Numerics

A

B

C

D

E

F

G

H

I

P

Q

R

S

T

U

V

W

X