Oracle® Universal Installer

Concepts Guide

Release 2.2

March 2002
Part No. A96697-01

ORACLE®

Contents

A   Sample Response File

B   Troubleshooting

Index

Send Us Your Comments

Oracle Universal Installer Concepts Guide, Release 2.2

Part No. A96697-01

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, please indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: nedc_doc@oracle.com
- FAX: 603-897-3317   Attn: Oracle Universal Installer - SMP Documentation
- Postal service:
  Oracle Corporation
  Oracle Universal Installer
  SMP Documentation
  One Oracle Drive
  Nashua, NH 03062
  USA

If you would like a reply, please give your name, address, telephone number, and (optionally) electronic mail address.

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This manual describes how to use the Oracle Universal Installer to install Oracle and third-party software.

Intended Audience

This manual is intended for users installing Oracle software products using the Oracle Universal Installer. This manual should be used in conjunction with any product-specific installation guides available with your Oracle software product.

Documentation Accessibility

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

Accessibility of Code Examples in Documentation   JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

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

Conventions

The following conventions are used in this manual:

Convention	Meaning
. . .	Vertical ellipsis points in an example mean that information not directly related to the example has been omitted.
. . .	Horizontal ellipsis points in statements or commands mean that parts of the statement or command not directly related to the example have been omitted
boldface text	Boldface type in text indicates a term defined in the text, the glossary, or in both locations.
< >	Angle brackets enclose user-supplied names.
[ ]	Brackets enclose optional clauses from which you can choose one or none.

1

Using the Oracle Universal Installer

Oracle Universal Installer is a Java application that can handle complex requirements. Oracle Universal Installer performs component-based installations and enables different levels of integrated bundle, suite, and Web-based installations, as well as complex logic in a single package. The installation engine is easily portable across all Java-enabled platforms, and platform-specific issues can be encapsulated from the overall installation process.

The Oracle Universal Installer provides the following capabilities for addressing software management and distribution:

- Automatic dependency resolution and complex logic handling
- Installation from the Web
- Component and suite installs
- Implicit de-install
- Support for multiple Oracle homes
- National Language Support (NLS)/globalization support
- Support for distributed installations
- Unattended "silent" installations using response files

> **Note:** Although the Oracle Universal Installer can list all products previously installed with another Installer (such as orainst), it cannot deinstall them.

New Features for Release 2.2

Oracle Universal Installer Release 2.2 offers the following new features:

- You can now record your Oracle Universal Installer session to a response file.

  This feature makes it easy to duplicate the results of a successful installation on multiple systems. All the options you selected during the installation are saved in the resulting response file. For information about recording your Universal Installer sessions see "Creating a Response File With Record Mode" on page 3-3.

- More accurate disk space calculations

  Oracle Universal Installer now uses a more accurate method of calculating the disk space required by your Oracle products. This feature reduces the risk of running out of disk space during an installation.

- Some Oracle products now take advantage of a new feature that allows the software to launch automatically immediately after the installation.

System Requirements

The Oracle Universal Installer system requirements are listed below:

Java Runtime Environment (JRE)
Automatically installed with the Oracle Universal Installer. Check the release notes or installation guide of the products you are installing for the exact version that is required.

Memory Requirements:
- Memory requirements vary depending on the number of components installed. Check the Release Notes of the products you are installing for details.

- 32 MB minimum recommended on Win32 computers

Disk Space Requirements
- 9 MB for Installer files

- 150 KB/component for other required files (for example, log files)

Installing Oracle Products

The following sections describe how to start the Oracle Universal Installer and install an Oracle product. Specifically, this section describes:

- Installing from a Single CD-ROM

- [Installing from Multiple CD-ROMs](#)
- [Installing Using a Response File](#)

> **Note:** If a later version of the Oracle Universal Installer is detected on your machine, the detected version is used to install rather than the one on the CD-ROM.

Installing from a Single CD-ROM

While installing Oracle Products contained in a single CD-ROM on the Windows NT platform, start the Oracle Universal Installer by running the Oracle Universal Installer executable, SETUP.EXE.

While installing Oracle Products residing in a single CD on UNIX platforms, run the script called `runInstaller` from the top directory of the CD-ROM by typing the following on the command line:

```
runInstaller
```

Installing from Multiple CD-ROMs

In the case of installing products that span multiple CDs for both UNIX and Windows platforms, temporary copies of OUI and JRE are placed in the TEMP or TMP directory so that these applications can be launched when you change CD-ROMs.

If you are having trouble installing a product from multiple CD-ROMs, try using the following procedure to unmount the first CD-ROM and mount the second CD-ROM. If you still have problems, refer to the documentation links at the end of this topic.

You may also need to launch the runInstaller script in the background by using the ampersand character:

```
./runInstaller &
```

By launching runInstaller in the background, you can change your current directory after you launch the Oracle Universal Installer so that you can eject the CD. You may want to create a shell script that launches the Oracle Universal Installer in the background and then exits. If you choose to create a shell script, remember to also pass all parameters that passed to the shell script along to `runInstaller` in case you want to pass a response file name to run a silent installation.

In most cases, the following procedure will help you with any problems you experience while switching to a second CD-ROM while you are installing an Oracle software. If you inadvertently run the installer while the current working directory is in the CD-ROM, follow these steps to mount the next CD-ROM:

1. Change directory to the root directory of your system and log in as the root user by using the following commands:

```
$ cd /
$ su root
```

2. Unmount and remove the CD-ROM from the CD-ROM drive using the following command:

```
# umount cdrom_mount_point_directory
```

3. Insert and mount the next CD-ROM into the CD-ROM drive by using the following command:

```
# mount options device_name cdrom_mount_point_directory
```

4. Enter the correct mount point in the Installation dialog box.

5. Click **OK** to continue.

If after attempting this procedure you are still having problems, refer to the section on installing from multiple CD-ROMS in the *Oracle9i Installation Guide*, which is available from the Oracle Documentation Center.

To locate documentation in the Oracle Documentation Center, enter the following URL in your Internet browser:

```
http://docs.oracle.com/
```

Specifically, use the following URL to display platform-specific information about mounting multiple CD-ROMS on a variety of UNIX systems:

```
http://docs.oracle.com/HTML_Storage/a90346/inst.htm#1089389
```

Installing Using a Response File

Many Oracle software products provide tools and procedures for running Oracle Universal Installer from the command line, without displaying the Universal Installer screens or responding to questions during the installation.

These kind of installations are called silent installations. Instead of prompting you to select a series of installation options, Universal Installer installs the software using a predefined set of options. These options are stored in an Oracle Universal Installer response file.

Consider the following information about Universal Installer response files:

- If your product installation includes a response file, you can find it on your stage (CD-ROM) under the `<root of CD>/response` directory.

- You can modify the response file for your Oracle product to customize an installation for your organization. For more information about customizing response files, see Chapter 3, "Customizing and Creating Response Files".

- You can create your own response files using the Oracle Universal Installer record mode. For more information, see Chapter 3, "Customizing and Creating Response Files".

- If you start Universal Installer programmatically from the command line, see "Using the Oracle Universal Installer Exit Codes" on page 1-14.

The following sections describe how to specify a response file when you start Oracle Universal Installer.

> **Note:** If you attempt to perform a silent installation on a UNIX computer where no Oracle products have been installed, you will receive an error message. Before you can perform a silent installation no such a computer, you must first run the script `oraInstRoot.sh`, which is saved in the `/tmp` directory. You must run this script with root privileges.

Specifying a Response File on Windows Systems

To start Oracle Universal Installer and specify the response file on Windows NT, enter the following command at the command line in the directory where the Universal Installer is installed:

```
setup.exe -responseFile <filename> <optional_parameters>
```

Specifying a Response File on UNIX Systems

On UNIX systems, enter the following at the command line in the directory where the Universal Installer is installed:

```
runInstaller -responseFile <filename> <optional_parameters>
```

> **Note:** You must specify the complete responseFile path.

> **Note:** Using the `-nowelcome` option with the `-silent` option is unnecessary since the Welcome screen does not appear when you use the `-silent` option.

Optional Parameters When Specifying a Response File

Optional parameters you may use with the `-responseFile` flag are:

- `-nowelcome`

- `-silent`

Use the `-nowelcome` flag along with the `-responseFile` flag to suppress the Welcome dialog that appears during installation.

Use the `-silent` flag along with the `-responseFile` flag to run the Universal Installer in complete silent mode. Note that the Welcome dialog is suppressed automatically.

Note that when the `-silent` flag is specified to the Universal Installer and if there are no values specified to a particular variable in a dialog, then the installer will abort. The success or failure of the installation, when this flag is specified, is generated in a file called `silentInstall.log`. This file will be generated in the `/tmp` directory in Solaris and the directory specified by the variable TEMP in Win32 platforms.

Using a Response File on Character Mode Consoles

When character mode consoles are used, Oracle recommends that you use the silent install, which is available with 8.1.5 and 8.1.6. Most (if not all) UNIX systems have the X packages by default. No additional installation is required.

In 8.1.5 and 8.1.6 versions of the Oracle Universal Installer, the Database Configuration Assistant and Net8 Configuration Assistant require that the X software runs. They all instantiate some GUI classes, but no GUI needs to be displayed if you running in silent mode. An installation can be run from a console (non-X-Window system or a telnet/dial up session) as long as the target system has X started.

Setting Response File Variables From the Command Line

Starting with Oracle Universal Installer 2.1, you can specify the value of certain variables when you start Universal Installer from the command line.

Specifically, you can specify session and component variables from the command line. For specific information about the format and organization of Universal Installer response files, see "Modifying a Response File" on page 3-2.

When you specify the value of a variable on the command line, that value overrides the value of the variable if it is defined in the response file.

Specifying the Value of a Session Variable To specify the value of a session variable on a UNIX system, use the following command syntax:

```
./runInstaller session_<variable_name>=<value>
```

For example, to prevent the Universal Welcome page from displaying:

```
./runInstaller session_SHOW_WELCOME_PAGE=false
```

To specify the value of a session variable on a Windows system, use the following command syntax:

```
setup.exe session_<variable_name>=<value>
```

For example, to prevent the Universal Welcome page from displaying:

```
setup.exe session_SHOW_WELCOME_PAGE=false
```

Specifying the Value of a Component Variable To specify the value of a component variable on a UNIX system, use the following command syntax:

```
./runInstaller <component_name>_<component_version>_<variable_name>=<value>
```
For example, to modify the value of a the variable VAR1 in version 1.1 of a component called COMP2:

```
./runInstaller COMP2_1.1_VAR1="test"
```

To specify the value of a component variable on a Windows system, use the following command syntax:

```
setup.exe <component_name>_<component_version>_<variable_name>=<value>
```

For example, to modify the value of a the variable VAR1 in version 1.1 of a component called COMP2:

```
setup.exe COMP2_1.1_VAR1="test"
```

Getting Help While Installing Oracle Products

At any time while installing your product, click **Help** for information about the screens specific to your installation.

Oracle Universal Installer provides two kinds of online help:

- Generic online help provided with every copy of Oracle Universal Installer

  These topics describe the screens and dialog boxes that every Oracle Universal Installer user sees, regardless of the product they are installing.

- Online Help specific to a particular installation

  These topics are created by the installation developer and describe the screens and dialog boxes specific to the product you are installing. For example, the help topic for the Installation Types page is often a custom help topic created by the installation developer that describes the specific installation types for the product you are installing.

After you display an online help topic, choose **Navigator** from the **Tools** menu to display the Navigator pane. From the navigator pane, you can browse a table the contents, select other topics, or search for a particular word or phrase in the Oracle Universal Installer online help.

Special Instructions for UNIX Users

The following sections describe special instructions that apply when you are installing certain products on a UNIX system.

Failed to Connect to Server Error

If you get an Xlib error or a "Failed to connect to Server" error when you are running Oracle Universal Installer on a Sun Solaris system, you need to define the following environment variables.

On the host computer where you are running Oracle Universal Installer, define the following variable:

```
%setenv DISPLAY <machine name>:0.0
```

Replace `<machine name>` with the name of the computer that will display Universal Installer.

On the computer that will display Universal Installer, enter the following command. This command allows other computers to display information on the computer's monitor:

```
%xhost +
```

Re-run the `runInstaller` script after you have set the DISPLAY environment variable.

Providing a UNIX Installer Location with Root Privileges

Various installation operations on the UNIX platform must be performed with root privileges. For example, you must have root privileges to be able to create the Oracle Universal Installer inventory.

If you do not have root privileges, you will be prompted to run a shell script from another terminal window before proceeding with the installation.

To successfully run this script:

1. Leave the Oracle Universal Installer window open and open another terminal window.

2. In the new terminal window, use the substitute user command to log in with root privileges:

```
su -root
```

3. Change directory to the Oracle Home into which you are currently installing your Oracle software product.

4. Run the shell script identified in the Oracle Universal Installer prompt.

5. When the script is finished and you are returned to the command prompt, exit from the new terminal window and return to Oracle Universal Installer to continue the installation.

Note: Do not exit the installation in order to run the shell script. Exiting the install removes this script.

You are prompted to run the script only the first time you install.

Providing a UNIX Group Name

If you are installing a product on a UNIX system, the Installer will also prompt you to provide the name of the group which should own the base directory.

You must choose a UNIX group name which will have permissions to update, install, and deinstall Oracle software. Members of this group must have write permissions to the base directory chosen.

Only users who belong to this group are able to install or deinstall software on this machine.

Oracle Universal Installer Concepts

The following sections describe:

- The Oracle Universal Installer inventory
- The Oracle Universal Installer ORAPARAM.INI file
- The Oracle Universal Installer Exit Codes

About the Oracle Universal Installer Inventory

The Oracle Universal Installer inventory is the location for the Oracle Universal Installer's bookkeeping. The inventory stores information about:

- All Oracle software products installed in all Oracle homes on a machine
- Other non-Oracle products, such as the Java Runtime Environment (JRE)

Starting with Oracle Universal Installer 2.1, the information in the Oracle Universal Installer inventory is stored in Extensible Markup Language (XML) format. The XML format allows for easier diagnosis of problems and faster loading of data. Any secure information is not stored directly in the inventory. As a result, during deinstallation of some products, you or your customer may be prompted for required secure information, such as passwords.

By default, the Universal Installer inventory is located in a series of directories at `/Program Files/Oracle/Inventory` on Windows computers and in the `<Oracle Home>/Inventory` directory on UNIX computers.

To identify the exact location of the inventory on your system, see the following sections.

Locating the Inventory on Windows systems

On the Windows platform, the Oracle Universal Installer first queries the registry to see if the following Oracle Home setting present:

```
HKEY_LOCAL_MACHINE
    Software
     Oracle
         INST_LOC
```

If this value is found, it will be used as the inventory location. If the value is not found, the following sequence occurs:

1. The Oracle Universal Installer requests the location of the `Program Files` directory.

   The Program Files folder is a system-defined directory usually on the C drive.

2. The directory structure is created in the form:

```
\<Program Files>
    \Oracle
        \bin
        \inventory
            \locks
            \logs
        \oui
```

3. The registry is updated with this inventory location.

Locating the Inventory on UNIX

At startup, the Oracle Universal Installer first looks for the following file: `var/opt/oracle/oraInst.loc`

If this value is found, it will be used as the inventory location. If the value is not found, the inventory path is created as follows:

`ORACLE_BASE\oraInventory`

If the ORACLE_BASE environment variable is not defined, the inventory is created at the same level as the first Oracle home. For example, if your first Oracle home is at `/private/ORACLEHome1`, then, the inventory is at `/private/oraInventory`.

If the Oracle Universal Installer is unable to create an inventory directory at these locations, it prompts you with a locations dialog.

Clicking the **Save As** button pops up a File browse dialog to accept a file name. The inventory as seen in the inventory tree will be logged into this file as text.

About the ORAPARAM.INI File

The ORAPARAM.INI file is the Oracle Universal Installer initialization file. This file includes information that defines the behavior of certain Universal Installer features. The installation developer provides a unique ORAPARAM.INI file for each product installation.

The ORAPARAM.INI file can help you troubleshoot problems and understand certain aspects of the Oracle Universal Installer product.

For example, for most installations, Universal Installer provides a default value on the File Locations page that points to the location of the product's installation kit or stage. This default value is stored in the ORAPARAM.INI file. The ORAPARAM.INI file also identifies the location of the Java Runtime Environment (JRE) required for the installation.

The ORAPARAM.INI file is located in the same directory as the command you use to start Oracle Universal Installer.

For detailed information about the contents of the ORAPARAM.INI file, see "Understanding the Contents of the ORAPARAM.INI File" on page B-4.

About Oracle Universal Installer Log Files

When you install or deinstall products using Oracle Universal Installer, important information about each install is saved not only in the Universal Installer inventory, but also in a series of log files. These log files can be used to troubleshoot installation problems. These files are also crucial for deinstalling and configuring the various software components you install on your Windows or UNIX computer.

For more information about the log files generated by Oracle Universal Installer, see the Oracle Universal Installer online help. For more information about using the Universal Installer online help, see "Getting Help While Installing Oracle Products" on page 1-8.

Running the Oracle Universal Installer After It is Installed

The Oracle Universal Installer is installed on your system during the installation of your Oracle products:

- On NT: `\Program Files\Oracle\oui`

- On Solaris: it is at the same level as `<orainventory>`

A new version of the Oracle Universal Installer replaces its older version.

The Oracle Universal Installer is placed under "Independent Products" in the Inventory panel as a "non-Oracle Home" product because it maintains an inventory across all Oracle Homes.

To run the Oracle Universal Installer:

- On Windows platforms, select **Start=>Programs=>Oracle Installation Products=>Oracle Universal Installer**.

- On Solaris, execute `./runInstaller` from the area where it is installed.

    For example: if the `<OraInventory>` is `/u01/oracle/oraInventory`; then, the Oracle Universal Installer will be at `/u01/oracle/oui`

When the Oracle Universal Installer is first run/installed, it checks for the JRE path (the location where it is run from), using the location specified in the `oraparam.ini` file. If the JRE does not exist here, the Oracle Universal Installer looks for a local JRE.

Determining the Default Oracle Home

By default, when you start Universal Installer, the software searches your system to determine the default Oracle Home where Oracle software should be installed.

The list of Oracle homes on Windows platforms consists of the following:

- all Oracle homes that are already created using the Oracle Universal Installer

- all homes created using the previous Oracle Installer (ORCA-based). Note: The Oracle Universal Installer only displays these homes. No actions can be performed on the homes creating using the previous Oracle Installer.

- the home that the ORACLE_HOME environment variable points to

The list of Oracle homes on UNIX consists of the following:

- all Oracle homes that are already created using the Oracle Universal Installer

- all homes as defined in the `/var/opt/oratab` file. Note: This location may vary for different UNIX platforms

- the home that the ORACLE_HOME environment variable points to

If none of these homes exists on a machine, a default home is calculated based on the disk volume with the largest amount of free space.

Using the Oracle Universal Installer Exit Codes

If you are starting and stopping Universal Installer programmatically (for example, by invoking Universal Installer using a response file), you may need to consider the exit codes generated by Universal Installer and perform a particular action depending on the code Universal Installer returns.

The Oracle Universal Installer returns one of the following exit codes:

Code	Description
0	All installations were successful
1	All installation were successful but some optional configuration tools failed
-1	At least one installation failed

Note that:

1. This feature will not work if the Oracle Universal Installer is running in "bootstrap" mode. In this case `setup.exe/runInstaller` will just launch the JRE process and return immediately without waiting for the exit code. The Oracle Universal Installer will be running in "bootstrap" mode if the following line exists in the file "oraparam.ini":

   `BOOTSTRAP=TRUE`

2. If you exit without installing any products (for example if you exit from the "Welcome" screen), the exit code will be -1.

Deinstalling Products

The following sections describe how to remove products you have installed with the Oracle Universal Installer.

Removing Oracle Products With the Oracle Universal Installer

A deinstallation can be performed before making your selection of products to install or after a successful install.

To remove Oracle products, perform the following steps:

1. Start the Oracle Universal Installer from a CD-ROM or

- For Windows platforms, launch the Oracle Universal Installer from the Start menu. Choose **Start > Installation Products > Oracle Universal Installer**.

- For UNIX platforms, run the script called `runInstaller` from the directory it was installed in, which is by default at the same level with the first Oracle Home called "oui" created on that machine.

  ```
  runInstaller
  ```

2. Click **Deinstall Products** on the "Welcome" screen.

   The Inventory panel appears.

3. Check the product(s) you want to remove from the Inventory panel and click **Remove**.

4. The Remove Confirmation Dialog appears, asking if you want to remove the products and their dependee components. Click **Yes**.

The Oracle Universal Installer warns you of any product dependencies that might cause problems if particular products are removed, and prompts you to confirm the de-installation.

Pay special attention to the full list of products being deinstalled before proceeding. The Oracle Universal Installer computes this list based on the dependencies of each component.

Note: You can also remove products by using the **Installed Products** button on the Oracle Universal Installer as long as this action is performed before making your selection of products to install.

Deinstalling Top Level Products that Have Dependees

A top level component is the most important component of an installation. It is the installable product you see at the first install screen. You can only install one top level component per install session.

The following examples use top level components that have dependees of different types offered by Oracle Universal Installer.

Test 1: Test Components Having the "required" Type of Dependees

Setup

Create 2 top level components (A, B), each having the same "required" dependee (C).

Install

I1a. Choosing A results in the install of A and C.

I1b. Later, choosing B results in only B being installed (its dependee C was previously installed).

Deinstall

D1a. Choosing A results in deinstall of A only. C is not deinstalled since it is also a "required" dependee of B, and thus is still needed.

D1b. Choosing A and C results in deinstalling A, B, and C. B is also deinstalled since its "required" dependee C is being deinstalled and it cannot be left installed without it.

D1c. Choosing C results in deinstalling A, B, and C. A and B are also deinstalled since their "required" dependee C is being deinstalled and they cannot be left installed without it.

Test 2: Test Components Having "1,more" Type of Dependees

Setup

Create 2 top level components (A, B), each having the SAME two '1,more' type dependees (C,D).

Install Test 2

I2.a. Choosing A and both dependees (C,D) results in the install of A,C,D.

I2.b. Later, choosing B (without C,D) results in the install of only B.

Deinstall Test 2

D2.a1. Choosing A results in deinstall of only A.

D2.a2. Choosing A and ONE dependee C (under A) results in deinstall of all (A,B,C,D). Note the deinstall of B and D occurs.

D2.a3. Choosing only ONE dependee C (under A) results in deinstall of all (A,B,C,D). Note the deinstall of A, B and D occurs.

D2.a4. Choosing only one dependee D (under A) results in deinstall of all (A,B,C,D). Note the deinstall of A, B and C occurs.

D2.a5. Choosing both dependee C and D (under A) results in deinstall of all (A,B,C,D). Note the deinstall of A and B occurs.

D2.b1. Choosing B results in deinstall of only B.

D2.b2. Choosing B and one dependee C (under B) results in deinstall of all (A,B,C,D). Note the deinstall of A and D occurs.

D2.b3. Choosing only one dependee C (under B) results in deinstall of all (A,B,C,D). Note the deinstall of A, B and D occurs.

D2.b4. Choosing only one dependee D (under B) results in deinstall of all (A,B,C,D). Note the deinstall of A, B and C occurs.

D2.b5. Choosing both dependee C and D (under B) results in deinstall of all (A,B,C,D). Note the deinstall of A and B occurs.

Forcing a Deinstall of a Component with a Different Internal Name

You cannot programmatically force the deinstallation of a certain component from a certain component; however, the Oracle Universal Installer's dependency engine should take care of the deinstallation should deinstall the component and replace it with the new component.

The rule is as follows:

If:

- A.v1 depends on B
- A.v2 depends on C
- A.v1 is in the replace range of A.v2

Then:

Upgrading A from v1 to v2 will perform the following:

- deinstall A.v1 and B
- install A.v2 and C

Note: Make sure the new version is marked to replace the old version. For example, the old version should be in the replace range of the new version.

> **Note:** Make sure the new version is marked to replace the old version. In the .prp, you can specify the version range to be replaced. For example, the old version should be in the replace range of the new version.

Silent Deinstallation

With Universal Installer you can not only perform silent installations, as described in "Installing Using a Response File" on page 1-4, but you can also perform silent deinstallations. A silent deinstallation allows you to remove Oracle applications from your system without using the Universal Installer graphical user interface.

You can choose to display no dialog boxes or prompts to the user, or you can selectively avoid displaying certain dialog boxes that are normally used during a deinstallation.

Immediately Displaying the Inventory Dialog Box

Use the following command on a Windows computer to immediately display the Inventory dialog box, which allows the user to select items for deinstallation without navigating the Universal Installer startup screen:

```
setup.exe –deinstall
```

On a UNIX computer, use the following command to produce the same result:

```
runInstaller –deinstall
```

Hiding the Inventory Dialog Box

If you would like to hide the inventory dialog box from the user during a deinstallation, you can specify the items to be deinstalled in the DEINSTALL_LIST parameter of the response file. For more information about response files, see "Installing Using a Response File" on page 1-4.

For information about the DEINSTALL_LIST parameter, see Chapter 3, "Customizing and Creating Response Files".

As with other response file parameters, you can also specify the DEINSTALL_LIST parameter on the Universal Installer command line. For example, on a Windows computer, enter:

```
runInstaller –deinstall DEINSTALL_LIST={"component1","1.0.1.2"}
```

For more information about using the command to specify response file parameters, see Chapter , "Setting Response File Variables From the Command Line".

Hiding the Deinstallation Confirmation and Progress Dialog Boxes

Use the following commands to hide the deinstallation confirmation and progress dialog boxes during a silent deinstallation on a Windows computer:

```
setup.exe -deinstall session_SHOW_DEINSTALL_PROGRESS=false
   session_SHOW_DESINSTALL_CONFIRMATION=false
```

Use the following commands to hide the deinstallation confirmation and progress dialog boxes during a silent deinstallation on a UNIX computer:

```
runInstaller -deinstall session_SHOW_DEINSTALL_PROGRESS=false
   session_SHOW_DESINSTALL_CONFIRMATION=false
```

Internationalization and Installation Translations Support

The following sections describe how Oracle Universal Installer supports multiple languages and internationalization.

Installer/Installation Dialogs Language

The Oracle Universal Installer runs in the language of the operating system. The Oracle Universal Installer uses the language that Java detects, the system locale value, and sets that to the default language.   The Installer dialogs will be displayed in this language if available. If specific Installer dialogs are not translated in the language of the operating system, those dialogs will be shown in English.

Oracle Universal Installer supports the following languages:

- German
- Latin American Spanish
- French
- Italian
- Japanese
- Korean
- Brazilian Portuguese

- Simplified Chinese
- Traditional Chinese

Product Language Selections

If multiple languages are defined for the products installed, the Universal Installer allows language selection via the Custom install type only.   The languages selections are for the top level component being installed. A top level component is the product selected on the Products Selection dialog, a single one per installation session. If any sub-components (lower level) do not have files defined for the language the user selects, English only will be installed. The English files are always installed for all components, regardless of user selection or operating system language.

If a component is already installed, but with a different language set, that component will be reinstalled.

At install time, the Oracle Universal Installer looks at the language list of each file/file group to determine which need to be copied.

2

Using Multiple Oracle Homes

Introduction

The Oracle Universal Installer supports the installation of several active Oracle homes on the same machine. An Oracle home is the system context in which Oracle products run. This context consists of the directory location where the products are installed, the corresponding system path setup, and where applicable, the program groups associated with the products installed in that home, and the services running from that home.

Multiple Oracle Homes

The Oracle Universal Installer supports the installation of several active Oracle homes on the same machine as long as the products support this at runtime. You can have multiple versions of the same product or different products running from different Oracle homes concurrently. Products installed in one home will not conflict or interact with products installed in another home. You can update software in any home at any time, assuming all Oracle applications/services/processes installed on the target home are shut down. Processes from other homes may still be running.

Target Home

The Oracle home currently accessed by the Oracle Universal Installer for installation or deinstallation is the target home. In order to upgrade or remove products from the target homes, those products must be shut down/stopped.

Product Types

Different levels of support are available with multiple Oracle homes for the following product types:

- Products that can be installed once in any new Oracle home. Ensure that the previous product installation is first removed so that only one copy/version of that application can run at the same time. If such product is already installed, and you try to install it again on a different home, the Oracle Universal Installer issues an error message.

- Products that can be installed multiple times, in different homes, on the same machine, with no conflicts or interaction at install-time, run-time, or deinstall-time. These are products which can be used concurrently.

> **Note:** The installation guide of your products will specify what type they are and whether they can be installed once in a home or multiple times.

Home Selector (Available on Win32 Platforms)

The following sections describe the Home Selector application, which is available when you install the Oracle Universal Installer on a Windows computer.

Overview

The Home Selector is part of the installation software and is installed with the Oracle Universal Installer. It is installed in `<oui_location>\HomeSelector`.

If your application is not installed in one of the Oracle homes, you need to ensure it uses the right product version, by pointing it to the right home. That specific (target) home must take precedence over the other homes in your path when you launch your application.

The Home Selector enables you to easily change your primary Oracle home (the one that appears first in the PATH environment variable). If you need to switch the active home or need to perform batch work which requires a "default home" to be active, the Oracle Home Selector can be used to change the Windows NT system settings.

When using the Home Selector to make a specific $ORACLE_HOME the active one, the software installation in question is moved to the top of the PATH variable, making it the first directory to be scanned for executables and library files.

How It Works

When you perform an install on a system, the first $ORACLE_HOME will be named the "DEFAULT_HOME" and will register itself in the Windows NT registry under the key:

```
HKEY_LOCAL_MACHINE
   Software
      Oracle
```

This is the default Windows NT registry hive which contains all the "generic" Oracle settings. Also the PATH variable is adjusted and the BIN directory of the $ORACLE_HOME is added to the environment variable.

Starting with Oracle 8.0.4, the first version to be multiple $ORACLE_HOME aware, some additional parameters will also be written to the key:

```
HKEY_LOCAL_MACHINE
   Software
      Oracle
         Home0
```

With the Oracle Universal Installer, you can change the "DEFAULT_HOME" name to something else. Also, it registers all Oracle settings in the "Home0" subkey of the ORACLE key.

When an additional $ORACLE_HOME is added to the system, the PATH variable is adjusted again to add the new BIN directory of the newly installed Oracle software. The registry variables are written to a key named "HOMEx," where X is the next available number in the Oracle key.

When several $ORACLE_HOMEs are installed, the executables referenced are all found in the first $ORACLE_HOME\BIN directory of the PATH variable. When you want to switch the order of the PATH variable, you can use the Home Selector to switch the order of the $ORACLE_HOMEs installed on the system.

Together with the installation and registration of the software in the registry, a second series of registry variables are written in the following key:

```
HKEY_LOCAL_MACHINE
   Software
      Oracle
         ALL_HOMES
```

This is the starting point for the list of all $ORACLE_HOME installations, done on this system.

The following variables are written in this key:

- HOME_COUNTER: Number of Oracle home directories already present on the system. This counter is a number, exactly 1 higher as the last ID subkey present.

- LAST_HOME: ID number of the currently active $ORACLE_HOME

- IDx: List of registry key's containing the name, number and installation directory of this ORACLE_HOME. The first installation is written in ID0, the next one in ID1, etc.

If the Home Selector is started in interactive mode, the registry hive "ALL_HOMES" is read, including all subkeys, and a dialog box will be displayed with all available installations. As soon as the you select one, the PATH variable will be adjusted at the system level.

In order to make batch processing possible, this tool also has a command line interface, making it possible to use the tool to switch the PATH settings while executing a series of commands:

```
ohsel -switch "Oracle Home Name"
```

The name of this $ORACLE_HOME is one of the names found in the ID subkeys.

Oracle Home Directory Structure for Windows Platform

A typical Oracle home on Windows platforms contains the files and directories shown in Table 2–1.

Table 2–1   Oracle Home Directory for Windows Platforms

Oracle Home Directory	Contents
\BIN	Product executables and DLLs
\DBS	Common message files
\PROD1	PROD1 product files
\PROD2	PROD2 product files

Optimal Flexible Architecture Directory Structure (on Solaris)

The Optimal Flexible Architecture (OFA) standard is a set of configuration guidelines for fast, reliable Oracle databases that require little maintenance.

OFA is designed to:

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

The OFA directory structure is described below.

ORACLE_BASE Directory

ORACLE_BASE specifies the BASE of the Oracle directory structure for OFA-compliant databases. A typical ORACLE_BASE directory structure is described in Table 2–2. When installing an OFA-compliant database using the Oracle Installer, ORACLE_BASE is by default set to */pm/app/oracle*.

Table 2–2   Sample ORACLE_BASE Directory Structure and Content

Directory	Content
admin	Administrative files
doc	online documentation
local	Subtree for local Oracle software
product	Oracle software

ORACLE_HOME Directory

The following is an example of an Oracle Server Install.

If you install an OFA-compliant Oracle Server, the ORACLE_HOME directory is /mount_point/app/oracle/product/*release_number*. ORACLE_HOME directory structure and content are described in Table 2–3. Under UNIX, the

ORACLE_HOME directory might contain the following subdirectories, as well as a subdirectory for each Oracle product selected.

> **Note:** The examples in this table are for sample only. The directories that appear in your own Oracle Home will depend upon whether the corresponding products are installed and the version of the Oracle database you are running.

Table 2–3   Sample ORACLE_HOME Directory Structure and Content

Directory	Content
assistants	configuration Assistants
bin	binaries for all products
ctx	interMedia Text cartridge
dbs	init*sid*.ora, lk*sid*
install	install related files
lib	Oracle product libraries
jlib	Java classes
md	Spatial cartridge
mlx	Xerox Stemmer (for interMedia Text cartridge)
network	Net8
nlsrtl	NLS runtime loadable data
ocommon	common files for all products
odg	data gatherer
opsm	Parallel Server Manager Components
oracore	core libraries
ord	data cartridges
otrace	Oracle TRACE
plsql	PL/SQL
precomp	precompilers
rdbms	server files and libraries required for the database

Table 2–3   Sample ORACLE_HOME Directory Structure and Content

Directory	Content
`slax`	SLAX parser
`sqlplus`	SQL*Plus

Creating an Oracle Home

To create an Oracle home, follow the steps below.

1. Run the Oracle Universal Installer.

2. In the File Locations page, enter the Oracle home settings for the installation session under Destination. See Table 2–4 for a description of the fields in this section of the screen.

Note: The Name field only appears in Windows platforms.

---

3. Continue with your installation. Refer to Chapter 1, "Using the Oracle Universal Installer" for detailed information.

Table 2–4   Oracle Installation Settings in Destination

Settings	Functions
Name	For Windows platforms, Oracle homes are identified by name, and the Oracle home name identifies the program group associated with a particular Oracle home, and the Oracle services installed on the associated home. The Oracle home name must be 1 to 16 characters long and can only include alphanumeric characters and underscores.

Table 2–4   Oracle Installation Settings in Destination (Cont.)

Settings	Functions
Path	Enter an Oracle home and its full path or select an Oracle home from a drop down list of existing Oracle Homes. The Oracle home location is the directory where products are installed.
	List of Oracle homes on Windows platforms:
	■ All Oracle homes that are already created using the Oracle Universal Installer
	■ All homes created using the previous Oracle Installer (ORCA-based)
	■ The home that the ORACLE_HOME environment variable points to
	List of Oracle homes on UNIX:
	■ All Oracle homes that are already created using the Oracle Universal Installer
	■ All homes as defined in the /var/opt/oratab file
	■ The home that the ORACLE_HOME environment variable points to
	If none of these homes exists, a default home is calculated based on the disk volume with the largest amount of free space.
	Datafiles may or may not be installed within an Oracle home. You may use the **Browse** button to choose a directory to install your product.
	For Windows platforms: it must be a valid path that is not in the Windows directory. Different homes cannot share the same location.

3

Customizing and Creating Response Files

This chapter introduces you to the concepts of silent installation and response files. It also describes how to modify or create a response file so you can customize and standardize the installation of Oracle products in your organization.

This chapter contains the following sections:

- About Response Files
- Modifying a Response File
- Creating a Response File With Record Mode
- Response File Format

About Response Files

This section answers the following questions:

- What Is a Silent Installation?
- What Is a Response File?
- Why Perform a Silent Installation?

What Is a Silent Installation?

A silent installation is an installation performed without displaying the Universal Installer screens.

Instead of prompting you to select a series of installation options, Universal Installer installs the software using a predefined set of options. These options are stored in an Oracle Universal Installer response file.

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 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 your product can be found on your stage (CD-ROM) under the `<root of CD>/response` directory.

Why Perform a Silent Installation?

Silent installations can be useful if you have to install an Oracle product multiple times on multiple computers. If the options you select while installing on each computer are always the same, 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 your Oracle products. This makes supporting those users easier 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 provided with your Oracle product.

> **Note:** If you attempt to perform a silent installation on a UNIX computer where no Oracle products have been installed, you will receive an error message. Before you can perform a silent installation no such a computer, you must first run the script `oraInstRoot.sh`, which is saved in the `/tmp` directory. You must run this script with root privileges.

The rest of this chapter describes the various sections and settings you can modify within an Oracle Universal Response file.

Modifying a Response File

If your product installation includes a response file template, you can find it on your stage (CD-ROM) under the `<root of CD>/response` directory.

If your product installation does not include a response file template, you can create a response file based on the installation options you select. For more information, see "Creating a Response File With Record Mode" on page 3-3.

To modify the response file:

1. Make a copy of the product's response file and open it in your favorite text editor.

2. Review any information provided in the response file or in the product installation guide.

   Many software products use settings in the response file to customize the installation of their particular product. Often, the product developers will provide you with suggestions or guidelines--possibly as notes within the response file--for modifying the response file provided with the software installation stage.

3. Get familiar with the organization and content of the response file using the information in the section "Response File Format" on page 3-5.

4. Modify the response file to meet the needs of your organization and save the modified version.

5. See the section "Installing Using a Response File" on page 1-4 for information on starting Universal Installer using your modified response file.

Creating a Response File With Record Mode

You can create a new response file, based on the installation options you select, by using Oracle Universal Installer Record mode.

When you use Record mode, Universal Installer records the installation session into a response file. You specify the name of the response file on the command line. The recorded response file is generated immediately after the Summary page, so you don't need to actually install your Oracle product to create the response file.

In other words, you can start the installation in Record mode and proceed through the installation options until you get to the Summary page. On the Summary Page, click **Exit** to stop the installation from proceeding with the installation. However, all the options you selected will be saved in the resulting response file.

You can use the newly created response file to run identical installation sessions on other computers in your organization.

Record mode can be also used during a silent installation. In those cases, the variable values specified in the original source response file will be recorded into the new response file.

The following sections describe how to use Universal Installer Record mode on Windows computers and on UNIX computers.

Using Record Mode on a Windows Computer

To create a new response on a Windows computer:

1. Open a Command Prompt window.

2. Change directory to the directory that contains setup file for your installation.

3. Enter the following command:

    ```
    setup –record –destinationFile <response_file_name>
    ```

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

    ```
    setup –record –destinationFile C:\response_files\install_oracle910
    ```

    Optionally, you can include the -silent option and create the response file during a silent installation.

4. Use the Universal Installer to select your installation options.

    When Universal Installer displays the Summary page, you can either continue with the installation or exit.

    Universal Installer saves your new response file the path and file name you specified on the command line.

Using Record Mode on a UNIX Computer

To create a new response on a UNIX computer:

1. Change directory to the directory that contains runInstaller script for your installation.

2. Enter the following command:

    ```
    ./runInstaller –record –destinationFile <response_file_name>
    ```

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

```
./runInstaller -record -destinationFile /private/temp/install_oracle91.rsp
```

Optionally, you can include the `-silent` option and create the response file during a silent installation.

3. Use the Universal Installer to select your installation options.

When Universal Installer displays the Summary page, you can either continue with the installation or exit.

Universal Installer saves your new response file using the location and file name you specified on the command line.

Response File Format

The following sections describe the organization and content of an Oracle Universal Installer response file.

How a Response File is Organized

Response files are divided into sections. Each section has a specific purpose and specific keywords and variables you can define.

> **Note:** Oracle Universal Installer treats wrong context, format, or type values within a response file as if no value were specified. In addition, variables which are outside of any section are ignored.

Sections of a Response File

Each section of a response file begins with a line, in brackets, that specifies the section name.

For example: `[GENERAL]` for the General Section, `[INCLUDE]` for the Include section, and `[Session]` for the Session section.

Response File Sections	Function
General	The General Section contains the version number of the response file.
Include	The Include section contains a list of response files that are included in this response file. Refer to the "Include Section of the Response File" on page 3-7 for more information.

Response File Sections	Function
Session	The Sessions section lists various dialogs of the Oracle Universal Installer.
Components	Component sections define public variables. They can also define installer variables.

Values Within a Section

Values for variables are specified as `<name> = <recommendation> : <value>`

The values that are given as `<value required>` must be specified for a silent installation to be successful.

To specify a default value, specify the value as `<name> = <default> : <value>`

For values that are given as `<value unspecified>`, you may optionally specify a value, where `<value>` can be one of the following types listed below:

Type	Represented As
Number	10
Boolean	true or false (either case can be used)
String	"Value"
String List	{"value1", "value2"}

`<recommendation>` can be specified as `Forced` or `Default`.

If you specify `Forced`, no dialog appears during installation. The value is automatically used. The user does not have the option to change the value.

If you specify `Default`, the dialog appears during installation with the value as the default. The user has the option to choose another value.

If nothing is specified for `<recommendation>`, `Forced` is used as the default.

For example of the format of values is shown below:
`InstallType=Default:"Minimum"`

Comments Within a Section

Comments begin with a "#" (hash or pound) symbol. They contain information about the type of the variable, state whether the variable appears in dialog, and

describe the function of the variable. A variable and a value are associated with a comment.

For example:

```
#TopLevelComponent;StringList;Used in Dialog
TopLevelComponent={"Demo", "1.0"}
```

Headers Within a Section

The header is a comment which describes the different formats and comment conventions used in a response file.

For example:

```
#Parameter : UNIX_GROUP_NAME
#Type : String
#Description : Unix group to be set for the inventory directory. Valid only in
Unix platforms.
#Example : UNIX_GROUP_NAME = "install"
UNIX_GROUP_NAME=<Value Unspecified>
```

General Section of the Response File

A General section appears in the beginning of a response file, providing the version number of the response file. There is only one General section per response file.

The General Section consists of the following variables

```
[GENERAL]
RESPONSEFILE_VERSION = <version_number>
```

Include Section of the Response File

The Include section is not automatically generated when you create a response file. If you want to include other response files in your main response file, you must add an Include section with a list of response files.

For example:

```
[INCLUDE]
FILE1="C:/Oracle/oracle/dba.rsp"
```

The values in the main response file take priority over the included response files.

> **Note:** Ensure that the [session] section is in the included file. The Oracle Universal Installer ignores any values that are assigned outside of a section.

Session Section of the Response File

The Sessions section lists pre-defined (generic) dialogs of the Oracle Universal Installer and shows whether the dialogs are visible to the user.

The dialog appears if the value is set to `true`. The dialog does not appear if its value is set to `false`.

An example is shown below:

```
[SESSION]
SHOW_INSTALL_PROGRESS_PAGE=true
```

The install progress page will be shown during installation.

The Sessions section also lists global variables that are set during one installation session (from the start of the installation to the end of installation). These variables include top level component, languages, and others.

Currently, the following variables are generated under the Sessions section.

FROM_LOCATION
"From" Location in the installer. The location of the source of the products to be installed.

Note: You must enter a value for FROM_LOCATION for a complete silent install.

FROM_LOCATION_CD_LABEL
FROM_LOCATION_CD_LABEL is used in multi-CD installations. It includes the label of the Compact Disk where the file "products.jar" exists. The label can be found in the file "disk.label" in the same directory as "products.jar".

LOCATION_FOR_DISK2
Complete path to other disks.

For example: `LOCATION_FOR_DISK2 = "..\..\Disk2"`

```
LOCATION_FOR_DISK2="F:\teststage\cd\Disk3"
```

If there are more than two disks, the above variable will be repeated as `LOCATION_FOR_DISK3` and so on.

The CD location for a silent installation is located by two mechanisms:

1. The Oracle Universal Installer looks for the Location_For_Disk{DiskNumber} variable in the response file and uses that location.

2. If the variable does not have a value or does not have the required files, it will look for the components under `../../Disk{DiskNumer}/stage` (from `products.jar`).

   Therefore, you can either specify the location to look for it in the Response file or you can copy them into the disk. For example:

   ```
   E:/Disk1/stage
   E:/Disk2/stage
   ```

NEXT_SESSION
Set this boolean variable to true if the installer needs to go to the File Locations page for another installation or to another response file if you are performing a silent installation.

Example: `NEXT_SESSION = true`

NEXT_SESSION_ON_FAIL
Set this boolean variable to true to allows users to invoke another installation session even if the the current installation session fails. This variable is used only if the NEXT_SESSION variable is set to true.

Example: `NEXT_SESSION_ON_FAIL = true`

NEXT_SESSION_RESPONSE
Use this string variable to enter the complete path of the next session's response file if you want to automatically begin another silent installation when the current installation is complete. If you specify only a file name, the Installer looks for the response file in the `<TEMP>/orainstall` directory. This variable is only used if NEXT_SESSION is set to true; otherwise, the Installer ignores the value of this variable.

Example: `NEXT_SESSION_RESPONSE="/private/usr2/nextinstall.rsp"`

ORACLE_HOME
"To" Location in the installer where products are to be installed.

> **Note:** You must enter a value for ORACLE_HOME for a complete silent install.

ORACLE_HOME_NAME
The name of the current Oracle home name.

Note: You must enter a value for ORACLE_HOME_NAME for a complete silent install.

APPL_TOP
The "To" location if you are installing into an Oracle Applications Top directory.

APPL_TOP_NAME
The name of the current Oracle Applications Top directory.

SHOW_COMPONENT_LOCATIONS_PAGE
The location page, which appears in a custom install type, can be suppressed by setting this value to false. If you set the value to false, the user is prevented from specifying alternate directories. If there are products with installed directories which can be changed, you may want to set the value to true.

SHOW_CUSTOM_TREE_PAGE
Set SHOW_CUSTOM_TREE_PAGE to true if the custom tree page in the installer must be shown. In the Custom Tree page, dependencies can be selected or unselected. This page appears only in a custom install type.

SHOW_END_SESSION_PAGE
The install success/failure page, which appears at the end of an installation, can be suppressed by setting this value to false.

SHOW_EXIT_CONFIRMATION
Set to true if the confirmation when exiting the installer needs to be shown.

Example: `SHOW_EXIT_CONFIRMATION = true`

SHOW_INSTALL_PROGRESS_PAGE
The install progress page, which appears during the install phase, can be suppressed by setting the value to false.

SHOW_OPTIONAL_CONFIG_TOOL_PAGE
Set to true if the Optional Configuration Tools page in the installer must be shown. The Optional Configuration Tools page shows the list of optional configuration tools that are part of this installation and the status of each tool, including detailed information on why the tool has failed.

SHOW_REQUIRED_CONFIG_TOOL_PAGE
Set to true if the Required Configuration Tools page in the installer must be shown. The Required Configuration Tools page shows the list of required configuration

tools that are part of the installation and the status of each tool, including detailed information on why the tool has failed.

SHOW_ROOTSH_CONFIRMATION
Set to true if the Confirmation dialog asking to run the root.sh script in the installer needs to be shown. This variable is valid only on Unix platforms.

Example: `SHOW_ROOTSH_CONFIRMATION = true`

SHOW_SPLASH_SCREEN
Set to true if the initial splash screen in the installer needs to be shown.

Example: `SHOW_SPLASH_SCREEN = true`

SHOW_SUMMARY_PAGE
The summary page can be suppressed by setting this value to false.

SHOW_WELCOME_PAGE
Set to true if the Welcome page in the installer needs to be shown.

Example: `SHOW_WELCOME_PAGE = false`

SHOW_RELEASE_NOTES
Set this parameter to true if you want the release notes for this installation to be shown at the end of the installation. A dialog box lists the available release notes. Note that the SHOW_END_SESSION parameter must be set to TRUE before you can use this parameter.

TOPLEVEL_COMPONENT
The name of the component (products) and the version as string list.

Note: You must enter a value for TOPLEVEL_COMPONENT.

Usually the components are represented with a pair of strings: the first one representing the internal name and the second representing the version.

For example, rdbms 8.1.5 is represented as, `{"oracle.rdbms","8.1.5.0.0}`.

UNIX_GROUP_NAME
The UNIX group name to be set for the inventory in UNIX platforms.

Note: The UNIX group name is used for first-time installations only.

DEINSTALL_LIST
Use this parameter to enter a list of components to be deinstalled during a silent deinstall session. For each component, specify the internal component name and version.

For example:

DEINSTALL_LIST={"sample1","1.0.0.0.0"}

SHOW_DEINSTALL_CONFIRMATION
Set this parameter to FALSE if you want to hide the deinstall confirmation dialog box during a silent deinstallation.

SHOW_DEINSTALL_PROGRESS
Set this parameter to FALS if you want to hide the deinstallation progress dialog box during a silent deinstallation.

Components Section of the Response File

The remaining sections in the response file are component sections, which are identified by a component's internal name and version.

In order to find a value for a variable, the Oracle Universal Installer looks under the appropriate component section to which the variable belongs.

A typical example of a component section is shown below:

```
[oracle.server_8.1.5.0.0]
```

> **Note:** Ensure that the `/var/opt/oracle/oraInst.loc` exists. The response file for Unix has a public variable, `setunixinstallgroup`, which only takes effect when it is the first Oracle Universal Installer installation and the `oraInst.loc` file is not already present in `/var/opt/oracle`.

Apart from the public variables defined in a component, each component can define the installer variables listed below:

- DEPENDENCY_LIST

- PROD_HOME

- OPTIONAL_CONFIG_TOOLS

- INSTALL_TYPE

- COMPONENT_LANGUAGES

DEPENDENCY_LIST
The DEPENDENCY_LIST is the list of dependees on which the component depends on. These dependee components represents the list of components that appear as "selected" during installation.

The DEPENDENCY_LIST variable is only generated when dependencies are present and if the dependency is not a required one.

You cannot list components to appear as de-selected items in a dialog.

Specifying required dependees is redundant since they will be selected anyway. The dependees selection can have the other two types of dependees: optional and at least one.

The list of components is specified by an internal name and version number.

A typical example is shown below:

```
DEPENDENCY_LIST={"oracle.netclt","8.1.5.0.0","oracle.netmgr","8.1.5.0.0"}
```

PROD_HOME

The PROD_HOME is the product location of the component on the target machine.

If you have an Oracle home component, the path specified must be in the Oracle home.

Note: The PROD_HOME variable is only used for non-Oracle Home components.

OPTIONAL_CONFIG_TOOLS

While all the required configuration tools are launched by the installer, you can control the configuration tools you would want to launch by specifying the tool's internal names in the OPTIONAL_CONFIG_TOOLS section.

Note: The OPTIONAL_CONFIG_TOOLS variable is only generated when at least one optional configuration tool is available.

You can specify both the Auto-launch optional tools and User-launch optional tools in a string list.

```
#Example : OPTIONAL_CONFIG_TOOLS = {"configtool2 ", "configtool3"}
OPTIONAL_CONFIG_TOOLS=<Value Unspecified>
```

If no value is specified for this variable, then all the tools are run by default.

If there is a value specified, then only those tools are run while the rest of the tools are ignored.

The way to suppress the configuration tool is to mention only the tools that you want to run as part of the OPTIONAL_CONFIG_TOOLS variable added per component. You should use the internal names of the configuration tool. The

response file generator generates these internal names also as part of the options given for the variable.

For example, oracle.server has tool1 and tool2 and you want to run only tool1 in the response file, you mention it as follows:

```
[oracle.server_81500]
OPTIONAL_CONFIG_TOOLS={"tool1"}
```

INSTALL_TYPE

You can set the install type variable to determine the install type of the currently selected top level component.

The install type variable is only generated for the top level components and only when there are more than one install type available.

Note: You must enter a value for INSTALL_TYPE.

COMPONENT_LANGUAGES

You can set the languages in which the components will be installed. You must use the internal name while specifying the value:

```
en,   : English
fr,   : French
de,   : German
ja,   : Japanese
```

For example: COMPONENT_LANGUAGES = {"en"}

Note: the COMPONENT_LANGUAGES variable is only generated when more than one is available.

A

Sample Response File

This chapter provides two samples that show how information is organized in a typical Oracle Universal Installer response file.

Example 1

```
##################################################################
## typical.rsp                                                  ##
## This is a sample responsefile for a typical EE install.      ##
##                                  (minimal)                   ##
## Copyright(c) Oracle Corporation 2000. All rights reserved.##
##                                                              ##
## Specify values for the variables listed below to customize##
## your installation.                                          ##
##                                                              ##
## Each variable is associated with a comment. The comment     ##
## identifies the variable type.                               ##
##                                                              ##
## Please specify the values in the following format:          ##
##                                                              ##
##          Type         Example                               ##
##          String       "Sample Value"                        ##
##          Boolean      True or False                         ##
##          Number       1000                                  ##
##          StringList   {"String value 1","String Value 2"}   ##
##                                                              ##
##                                                              ##
##          !!!!!IMPORTANT!!!!!                                 ##
##                                                              ##
##   PLEASE VERIFY (CUSTOMIZE) THE VALUES FOR THE FOLLOWING     ##
##   VARIABLES BEFORE USING THIS FILE FOR SILENT INSTALL.TKX!   ##
##                                                              ##
```

Example 1

```
##  1) UNIX_GROUP_NAME                                       ##
##  2) ORACLE_HOME && ORACLE_HOME_NAME                       ##
##  3) s_GlobalDBName                                        ##
##  4) s_dbSid                                               ##
##  5) s_mountPoint                                          ##
##                                                           ##
##  ALSO, MAKE SURE /var/opt/oracle/oraInst.loc EXISTS.      ##
##  TYPICAL ENTRY IN "oraInst.loc" LOOKS LIKE -              ##
##  inventory_loc=/home/dba/oraInventory                     ##
##  BASICALLY, WHEREVER INVENTORY NEEDS TO BE LOCATED.       ##
##                                                           ##
##  AT THE END OF SILENT INSTALL LOOK UP THE FOLLOWING LOGS: ##
##  - oraInventory/logs/installActions.log                   ##
##     (for $ORACLE_HOME/root.sh execution, etc.)            ##
##  - /tmp/silentInstall.log                                 ##
##                                                           ##
###############################################################

[General]


RESPONSEFILE_VERSION=1.0.0.0.0

[Session]


#This entry is not used on Windows platforms
#UNIX_GROUP_NAME;String;Used in Dialog
#Group that the current user is part of, for ownership of the install files.
#UNIX_GROUP_NAME="dba"


#FROM_LOCATION;String;Used in Dialog
#Full path for the products.jar file.
#Replace the X: with the drive letter of your CD-ROM device.
FROM_LOCATION="X:\stage\products.jar"


#FROM_LOCATION_CD_LABEL;String:Used in multi-CD installations.
#It includes the label of the Compact Disk where the file "products.jar" exists.
#The label can be found in the file "disk.label" in the same directory as
#"products.jar".


#ORACLE_HOME;String;Used in Dialog
#Enter the path to your oracle home.  Replace the drive letter and optionally
#alter the home path.
ORACLE_HOME="X:\Oracle\Ora81"


#ORACLE_HOME_NAME;String;Used in Dialog
```

Example 1

```
#Enter the name of this oracle home.  The name will be used to identify this
#home.
ORACLE_HOME_NAME="OraHome81"

#TOPLEVEL_COMPONENT;StringList;Used in Dialog
#Choices: "oracle.server", "8.1.5.0.0"
TOPLEVEL_COMPONENT={"oracle.server", "8.1.5.0.0"}

#SHOW_COMPONENT_LOCATIONS_PAGE;Boolean;Used in Dialog
#Some components are flexible in where they are installed, although all have
# default locations. Set this to FALSE if you want to accept the default
locations
# and not show this page.
SHOW_COMPONENT_LOCATIONS_PAGE=false

#SHOW_SUMMARY_PAGE;Boolean;Used in Dialog <--------------
#MUST be <false> for "silent" install
SHOW_SUMMARY_PAGE=false

#SHOW_INSTALL_PROGRESS_PAGE;Boolean;Used in Dialog <--------------
#MUST be <false> for "silent" install
SHOW_INSTALL_PROGRESS_PAGE=false

#SHOW_REQUIRED_CONFIG_TOOL_PAGE;Boolean;Used in Dialog
SHOW_REQUIRED_CONFIG_TOOL_PAGE=false

#SHOW_OPTIONAL_CONFIG_TOOL_PAGE;Boolean;Used in Dialog
SHOW_OPTIONAL_CONFIG_TOOL_PAGE=false

#SHOW_END_SESSION_PAGE;Boolean;Used in Dialog <--------------
#MUST be <false> for "silent" install
SHOW_END_SESSION_PAGE=false

#Parameter : LOCATION_FOR_DISK2
#Type : String
#Description : Complete path to the other disks.
#Example : LOCATION_FOR_DISK2 = "..\..\Disk2"
LOCATION_FOR_DISK2="F:\teststage\cd\Disk3"

#Parameter : NEXT_SESSION
#Type : Boolean
#Description : Set to true if the installer needs to go to the File
Locations page for another installation.
#Example : NEXT_SESSION = true
NEXT_SESSION=true
```

Example 1

```
#Parameter : SHOW_SPLASH_SCREEN
#Type : Boolean
#Description : Set to true if the initial splash screen in the installer
needs to be shown.
#Example : SHOW_SPLASH_SCREEN = true
SHOW_SPLASH_SCREEN=true

#Parameter : SHOW_WELCOME_PAGE
#Type : Boolean
#Description : Set to true if the Welcome page in the installer needs to
be shown.
#Example : SHOW_WELCOME_PAGE = false
SHOW_WELCOME_PAGE=false

#Parameter : SHOW_ROOTSH_CONFIRMATION
#Type : Boolean
#Description : Set to true if the Confirmation dialog asking to run the
root.sh script in the installer needs to be shown.
#Valid only in Unix platforms.
#Example : SHOW_ROOTSH_CONFIRMATION = true
SHOW_ROOTSH_CONFIRMATION=true

#Parameter : SHOW_EXIT_CONFIRMATION
#Type : Boolean
#Description : Set to true if the confirmation when exiting the
installer needs to be shown.
#Example : SHOW_EXIT_CONFIRMATION = true
SHOW_EXIT_CONFIRMATION=true

[oracle.server_8.1.5.0.0]
#INSTALL_TYPE;String;Used in Dialog
# Minimal - Optional pre-configured DB (b_createDB), Networking services, Server
utilities
# Typical - Pre-configured starter DB, Licensable options and Cartridges,
#          Networking services, Server utilities

INSTALL_TYPE="Typical"

[oracle.rdbms_8.1.5.0.0]

#OPTIONAL_CONFIG_TOOLS;StringList
#OPTIONAL_CONFIG_TOOLS={"dbassist"}  OR OPTIONAL_CONFIG_TOOLS={"Oracle Database
Configuration Assistant"}

#s_GlobalDBName;String;Used in Dialog
```

Example 1

```
#This is the name of the database which will be created as part of the typical
install.
#You should modify this string to use an appropriate name.
s_GlobalDBName="acme.us.oracle.com"

#This entry not needed for Windows installation.
#s_mountPoint;String;Used in Dialog
#Database file location: directory for datafiles, control files, redo logs
#s_mountPoint="/home/oracle/dbs"

#s_dbSid;String;Used in Dialog
#Value for ORACLE_SID. Change this to an appropriate SID.
s_dbSid="ORCL"

#b_createDB;Boolean
#Relevant only in MINIMAL, set to TRUE if you want a starter database, set to
FALSE if not.
#b_createDB=true
```

Example 2

Example 2

```
####################################################################
## typical.rsp                                                    ##
## This is a sample responsefile for a typical EE install.        ##
##                                     (minimal)                  ##
## Copyright(c) Oracle Corporation 1998. All rights reserved.##
##                                                                ##
## Specify values for the variables listed below to customize##
## your installation.                                            ##
##                                                                ##
## Each variable is associated with a comment. The comment        ##
## identifies the variable type.                                  ##
##                                                                ##
## Please specify the values in the following format:             ##
##                                                                ##
##        Type         Example                                    ##
##        String       "Sample Value"                             ##
##        Boolean      True or False                              ##
##        Number       1000                                       ##
##        StringList   {"String value 1","String Value 2"}  ##
##                                                                ##
##                                                                ##
##          !!!!!IMPORTANT!!!!!                                   ##
##                                                                ##
##  PLEASE VERIFY (CUSTOMISE) THE VALUES FOR THE FOLLOWING        ##
##  VARIABLES BEFORE USING THIS FILE FOR SILENT INSTALL.TKX! ##
##                                                                ##
##  1) UNIX_GROUP_NAME                                            ##
##  2) ORACLE_HOME && ORACLE_HOME_NAME                            ##
##  3) s_GlobalDBName                                             ##
##  4) s_dbSid                                                    ##
##  5) s_mountPoint                                               ##
##                                                                ##
##  ALSO, MAKE SURE /var/opt/oracle/oraInst.loc EXISTS.           ##
##  TYPICAL ENTRY IN "oraInst.loc" LOOKS LIKE -                   ##
##  inventory_loc=/home/dba/oraInventory                          ##
##  BASICALLY, WHEREVER INVENTORY NEEDS TO BE LOCATED.            ##
##                                                                ##
##  AT THE END OF SILENT INSTALL LOOK UP THE FOLLOWING LOGS: ##
##  - oraInventory/logs/installActions.log                        ##
##    (for $ORACLE_HOME/root.sh execution, etc.)                  ##
##  - /tmp/silentInstall.log                                      ##
##                                                                ##
####################################################################
```

Example 2

```
[General]

RESPONSEFILE_VERSION=1.0.0.0.0

[Session]

#This entry is not used on Windows platforms
#UNIX_GROUP_NAME;String;Used in Dialog
#Group that the current user is part of, for ownership of the install files.
#UNIX_GROUP_NAME="dba"

#FROM_LOCATION;String;Used in Dialog
#Full path for the products.jar file.
#Replace the X: with the drive letter of your CD-ROM device.
FROM_LOCATION="G:\81500\EEdition\990211_CDM2\stage\products.jar"

#ORACLE_HOME;String;Used in Dialog
#Enter the path to your oracle home.  Replace the drive letter and optionally
#alter the home path.
ORACLE_HOME="E:\16080_2.12\815Sil"

#ORACLE_HOME_NAME;String;Used in Dialog
#Enter the name of this oracle home.  The name will be used to identify this
#home.
ORACLE_HOME_NAME="OraHome81"

#TOPLEVEL_COMPONENT;StringList;Used in Dialog
#Choices: "oracle.server", "8.1.5.0.0"
TOPLEVEL_COMPONENT={"oracle.server", "8.1.5.0.0"}

#SHOW_COMPONENT_LOCATIONS_PAGE;Boolean;Used in Dialog
#Some components are flexible in where they are installed, although all have
# default locations. Set this to FALSE if you want to accept the default
locations
# and not show this page.
SHOW_COMPONENT_LOCATIONS_PAGE=false

#SHOW_SUMMARY_PAGE;Boolean;Used in Dialog <---------------
#MUST be <false> for "silent" install
SHOW_SUMMARY_PAGE=false

#SHOW_INSTALL_PROGRESS_PAGE;Boolean;Used in Dialog <---------------
#MUST be <false> for "silent" install
SHOW_INSTALL_PROGRESS_PAGE=false
```

Example 2

```
#SHOW_REQUIRED_CONFIG_TOOL_PAGE;Boolean;Used in Dialog
SHOW_REQUIRED_CONFIG_TOOL_PAGE=false

#SHOW_OPTIONAL_CONFIG_TOOL_PAGE;Boolean;Used in Dialog
SHOW_OPTIONAL_CONFIG_TOOL_PAGE=false

#SHOW_END_SESSION_PAGE;Boolean;Used in Dialog <---------------
#MUST be <false> for "silent" install
SHOW_END_SESSION_PAGE=false

[oracle.server_8.1.5.0.0]
#INSTALL_TYPE;String;Used in Dialog
# Minimal - Optional pre-configured DB (b_createDB), Networking services, Server
utilities
# Typical - Pre-configured starter DB, Licensable options and Cartridges,
#          Networking services, Server utilities

INSTALL_TYPE="Typical"

[oracle.rdbms_8.1.5.0.0]

#OPTIONAL_CONFIG_TOOLS;StringList
#OPTIONAL_CONFIG_TOOLS={"dbassist"}  OR OPTIONAL_CONFIG_TOOLS={"Oracle Database
Configuration Assistant"}

#s_GlobalDBName;String;Used in Dialog
#This is the name of the database which will be created as part of the typical
install.
#You should modify this string to use an appropriate name.
s_GlobalDBName="B815.us.oracle.com"

#This entry not needed for Windows installation.
#s_mountPoint;String;Used in Dialog
#Database file location: directory for datafiles, control files, redo logs
#s_mountPoint="/home/oracle/dbs"

#s_dbSid;String;Used in Dialog
#Value for ORACLE_SID. Change this to an appropriate SID.
s_dbSid="B815"

#b_createDB;Boolean
#Relevant only in MINIMAL, set to TRUE if you want a starter database, set to
FALSE if not.
b_createDB=true
```

B

Troubleshooting

This chapter describes possible troubleshooting issues.

- Debugging Mechanisms in the Oracle Universal Installer
- Oracle Universal Installer Errors
- Deinstalling or Upgrading Oracle Universal Installer
- Understanding the Contents of the ORAPARAM.INI File
- Other Tips

Debugging Mechanisms in the Oracle Universal Installer

During the install, the Oracle Universal Installer writes a text file that contains information on:

- variable settings
- action calls
- queries
- exception information

The `installActions.log` is located in the `<oraInventory>\logs` directory. It is backed-up as `installActions<date>.log`.

The Installation Log is used for debugging purposes only. All actions, queries, everything that happens during the install, and all modifications to the target machine are logged in the file.

A sample file is shown below:

```
installActions1999-06-16_09-52-16-AM.log
```

```
Environment variables:
ORACLE_HOME =
PATH = E:\OEM\bin;E:\ORACLE815\bin;C:\Program
Files\Oracle\jre\1.1.7\bin;C:\ORATST\BIN
CLASSPATH = C:\Program Files\Exceed.nt\hcljrcsv.zip;
Username is: pjlee

The installer version is 1.7.0.0.0

*** Welcome Page***
Setting value of FROM_LOCATION to E:\bootstrap\cd\Disk1\stage\products.jar
Setting value of INVENTORY_LOCATION to C:\Program Files\Oracle\Inventory
Setting value of UNIX_GROUP_NAME to
Setting value of FROM_LOCATION to E:\bootstrap\cd\Disk1\stage\products.jar
Setting value of ORACLE_HOME to E:\Universal
Setting value of ORACLE_HOME_NAME to Universal

*** File Locations Page***
FromLocation = FROM_LOCATION = E:\bootstrap\cd\Disk1\stage\products.jar
ToLocation = ORACLE_HOME = E:\Universal
ToName = ORACLE_HOME_NAME = Universal
Initializing installer access setup
Setting value of ORACLE_HOME to E:\Universal
Setting value of ORACLE_HOME_KEY to Software\ORACLE\HOME3
Setting value of ORACLE_HOME_FOLDER to Oracle - Universal
Setting value of ORACLE_HOME_SERVICE to Universal
Setting value of ToplevelComp to oracle.swd, 1.7.0.0.0, >0.0, [ 912 453 615 50
601 173 467 295 87 610 198 918 913 162 2 30 21 10021 111 90 168 888 ][OH:0]
Setting value of SELECTED_LANGUAGES to [en]
Doing operation for installer access setup
Initializing inventory setup WCCE
Doing operation in inventory setup WCCE

*** Available Products Page***
TopLevelComp = ToplevelComp = oracle.swd, 1.7.0.0.0, >0.0, [ 912 453 615 50 601
173 467 295 87 610 198 918 913 162 2 30 21 10021 111 90 168 888 ][OH:0]
LangsSel = SELECTED_LANGUAGES = [en]
Setting value of ToplevelComp to oracle.swd, 1.7.0.0.0, >0.0, [ 912 453 615 50
601 173 467 295 87 610 198 918 913 162 2 30 21 10021 111 90 168 888 ][OH:0]
Setting value of DepMode to Complete
Setting value of TLDepModes to Complete,

*** Installation Types Page***
DepMode = DepMode = Complete
TLDepModes = TLDepModes = Complete,
```

```
Setting value of DepMode to Complete
Setting value of PROD_HOME to E:\Universal\oracle.swd
Setting value of PROD_HOME to E:\Universal\oracle.swd
Setting value of PRE_REQUISITE to true
Setting value of PROD_HOME to E:\Universal\oracle.swd
Setting value of SHOW_COMPONENT_LOCATIONS_PAGE to true

*** Component Locations Page***
ShowCompLocs = SHOW_COMPONENT_LOCATIONS_PAGE = true
Entering  component: oracle.swd installation

*** Summary Page***
ShowSummary = SHOW_SUMMARY_PAGE = null
Global Settings
Source : E:\bootstrap\cd\Disk1\stage\products.jar
Destination : E:\Universal
```

Text files are also written in the `temp` directory:

```
<temp>\OraInstall\oraInstall.err
<temp>\OraInstall\oraInstall.out
```

If a problem occurs during the installation, you should look for Java exceptions in these files.

Oracle Universal Installer Errors

Refer to the Release Notes for information on any Oracle Universal Installer limitations for a particular version.

The most common Oracle Universal Installer errors are listed below:

- out of "temp" space

  Make sure you have enough space in the default TEMP or TMP directory otherwise the installation will not succeed. The amount of space required depends on the product being installed.

- incomplete stage and missing files in file groups

  Verify the staging file. There may be runtime errors that the install developer missed which causes the unexpected environment changes.

- exceptions from action libraries:

  - NT Services, if a service is already started

- file permissions

- UNCAUGHT_RUNTIME exception when user tries to install the product on his machine

  This is a blanket error catch which catches any runtime errors like NullPointers from libraries. Check if the temp\OraInstall\orainsrall.err has any stack trace. If there is none, the user will have to call support to debug his staging area.

Deinstalling or Upgrading Oracle Universal Installer

Consider the following restrictions and guidelines when deinstalling or upgrading Oracle Universal Installer:

- You cannot deinstall the Oracle Universal Installer if you are running it from the installed area. To deinstall the Oracle Universal Installer, run it from the stage area.

- To upgrade the Oracle Universal Installer, you need to run it from the staging area.

- To preserve backward compatibility, Oracle Universal Installer 2.1 can read serialized inventories written by Universal Installer 2.0.x and older. As a result, Universal Installer 2.1 will always be able to read the changes made by an older version of Universal Installer.

Understanding the Contents of the ORAPARAM.INI File

The following sections describe the contents of the ORAPARAM.INI file, which is used to set initialization parameters for Oracle Software Installer. In general, you should not have to edit the contents of this file, but in certain situations, understanding the contents of this file can help you troubleshoot installation problems.

The following sections include:

- A description of the parameters that can be used in the ORAPARAM.INI file

- A sample ORAPARAM.INI file

Parameters in ORAPARAM.INI

This section describes the various parameters that can be set in the ORAPARAM.INI file.

[Oracle]
DISTRIBUTION
Set DISTRIBUTION to TRUE if the ORAPARAM.INI file is on a distribution media. A distribution media can be a CD-ROM or the Web.

SOURCE
Location of your staging area (the `products.jar` file). This location is relative to the directory where `oraparam.ini` exists.

LICENSE_LOCATION
Location of a text file for License information. This location is relative to the directory where oraparam.ini exists. The legal terms file should be a plain text file.

If you specify the LICENSE_LOCATION variable, the Oracle Universal Installer will ask for license acceptance after you click the Next button on the "Welcome" screen. The Oracle Universal Installer will let the user proceed only after license is accepted.

This parameter is ignored if the file is not found.

LICENSE_TITLE
The value of this variable displays as the title of the license agreement. The Oracle Universal Installer only reads this value if the license dialog is displayed. That is, if the LICENSE_LOCATION variable has a valid value.

JRE_LOCATION
Location to the Java Runtime Environment (JRE) which will be used by the Oracle Universal Installer.

Note: For a single install to point to two different JRE versions, move the oraparam.ini file one level below its original location and then specify the individual platform's JRE location. For example, if the original location is in the Disk1/install directory, move it to the Disk1/install/win32 or the Disk1/install/solaris directory and then specify the individual platform's JRE location.

OUI_LOCATION
Location of the Oracle Universal Installer files.

JRE_MEMORY_OPTIONS
Set these to increase the initial heap size for JRE.

DEFAULT_HOME_LOCATION
Location of the default home.

DEFAULT_HOME_NAME
The default name for the Oracle Home. This parameter is used only if the install takes place on a machine with no previous Oracle installations.

NO_BROWSE
List directories which you do not want to browse. These are typically large directories which will take a long time to list. For example, `/net`, `/nfs`.

NLS_ENABLED
Set NLS_ENABLED to TRUE for the Oracle Universal Installer to enable NLS support. Set NLS_ENABLED to FALSE to disable the installation session translations. The Oracle Universal Installer displays in English even if you run on a non-English system.

BOOTSTRAP
In the case of installing products that span multiple CDs, a "bootstrap" occurs in which temporary copies of OUI and JRE are placed in the TEMP or TMP directory so that these applications can be launched when you change CD-ROMs.

As a user or developer of an install, you need to set BOOTSTRAP to True or False. BOOTSTRAP tells the Oracle Universal Installer to attempt a bootstrap. Set BOOTSTRAP to True before cutting CDs, but set it to False once you have copied the staging area to the hard disk.

If you are running a multiple CD install, you must set the bootstrap variable BOOTSTRAP=TRUE.

Otherwise, the first CD cannot be ejected on Solaris because the file system is busy. The Oracle Universal Installer is holding open several files on the CD including the JRE executable, and on Solaris, ejectable volumes cannot be ejected if a process has opened files on the volume.

Setting the BOOTSTRAP variable to TRUE informs the Oracle Universal Installer to copy the Oracle Universal Installer classes and the JRE executables to a temporary directory where you can launch them; thereby freeing the CD device.

You may also need to launch runInstaller in the background by using the ampersand character:

```
./runInstaller &
```

By launching `runInstaller` in the background, you can change your current directory after you launch the Oracle Universal Installer so that you can eject the CD.

You may want to create a shell script that launches the Oracle Universal Installer in the background and then exits. If you choose to create a shell script, remember to also pass all parameters that passed to the shell script along to runInstaller in case you want to pass a response file name to run a silent installation.

BOOTSTRAP_SIZE

Use this parameter to set the size the temporary space required when you the BOOTSTRAP parameter is set to TRUE.

For example, when you install Oracle Universal Installer, Universal Installer sets the value of this parameter to to the temporary space required by both Universal Installer and the Java Runtime Environment (JRE).

By default, if the BOOTSTRAP_SIZE entry is not set, Universal Installer will assume 30MB for Windows computers and 54MB for UNIX computers. However, this value could vary from one major release to the other, based on the space required by newer versions of JRE.

If the shiphome contains advertisement images, the installation developers are expected to add the space taken by the images to this value.

OUI will check the temp space requirements before starting up and give an error if there is not enough space for OUI to run in bootstrap mode.

OUI_VERSION = <2.0.1.0.0>

Set the version of the Oracle Universal Installer you are using.

> **Note:** You must set the Universal Installer version correctly in order for the BOOTSTRAP parameter to work.

USE_BUILD_NUMBER

Use this parameter to control whether or not Universal Installer considers the build number of the component when determining whether or not to overwrite a previous version or the same version of a component.

APPLTOP_STAGE

Use this parameter when you want your users to specify the location of an Oracle Applications top (APPL_TOP) directory. When this parameter is set to true, the File Locations page will include fields for selecting an APPL_TOP directory.

[IMAGES]

To show advertisements during an install, you must specify each image as a separate variable.

Sample ORAPARAM.INI

A sample ORAPARAM.INI file is shown below.

```
[Oracle]
DISTRIBUTION=TRUE
#Set DISTRIBUTION to TRUE if the ORAPARAM.INI file is on a distribution media. A
distribution media can be a CD-ROM or the Web.

SOURCE=../stage/products.jar
#Location of the products.jar file. This location is relative to the directory
where oraparam.ini exists.

LICENSE_LOCATION=
#Location of a text file for License information. This location is relative
to the directory where oraparam.ini exists.
#After you click the Next button on the Welcome page, the Oracle Universal
Installer will ask for license acceptance. If this file exists, the Oracle
Universal Installer will let the user proceed only after license is accepted.

JRE_LOCATION=../stage/Components/oracle/swd/jre/1.1.6.2.1o/2/DataFiles/Expanded
/ojre
#Relative location to the Java Runtime Environment (JRE) which will be used by
the Oracle Universal Installer

OUI_LOCATION=../stage/Components/oracle/swd/oui/1.5.1.4.2/1/DataFiles/Expanded
#Relative location to the Oracle Universal Installer files

DEFAULT_HOME_NAME="OUIHome"
#The default name for the Oracle Home. This parameter is used only if the
install takes place on a machine with no previous Oracle installations. This
parameter is used only on Windows platforms.

HLP_LOCATION=
#For future use only

EWT_LOCATION=
#For future use only

NLS_ENABLED=TRUE
#Set NLS_ENABLED to FALSE to disable the translations. The Oracle Universal
Installer displays in English even if you run on a non-English system.

JRE_MEMORY_OPTIONS=-ms16m -mx32m
#Set these to increase the initial heap size for JRE.
```

```
NO_BROWSE=
#list directories which you do not want to browse. These are typically
large directories which will take a long time to list.
# =/net, /nfs

OUI_VERSION = <1.7.0.1.0>
Set the version of the Oracle Universal Installer you are using. Note: you must
set the version correctly in order for the bootstrap to work.

USE_BUILD_NUMBER=True
When this parameter is set to true, Oracle Universal Installer will consider the
build number of the component when determining whether or not to overwrite a
previous version or the same version of a component.

APPLTOP_STAGE=True
Use this parameter when you want your users to specify the location of an Oracle
Applications top (APPL_TOP) directory. When this parameter is set to true, the
File Locations page will include fields for selecting an APPL_TOP directory.


[IMAGES]
FILE1=images/installAd0.gif
FILE2=images/installAd1.gif
```

Other Tips

Make sure the correct version of the JRE is used from the PATH.

If Java cannot run applications from a Sun machine on an NCD X terminal (the Oracle Universal Installer does not start), remove the following file:

```
$JAVA_HOME/lib/font.properties
```

X terminal emulators which were tested with the Oracle Universal Installer are listed below:

- Exceed

- Reflection X

  Refection X 6.2+: Select Settings/Window Manager/ Default Local Window Manager/ Reflection Window Manager. You may also have to select "X terminal desktop."

Reflection X 6.0: Select Connection/New XDMCP Connection/Connection Settings/Direct method/ type host destination and press connect.

If you are deleting an Oracle Home manually without using the Oracle Universal Installer, the products remain registered with the Oracle Universal Installer. You must then simulate a de-install so that all Oracle Universal Installer references are removed.

If the install is canceled:

- Products will not be registered with the Oracle Universal Installer

- Some files may have been copied, depending on when you issued the Cancel.

Index