Oracle Tuxedo Application Rehosting Workbench

Installation Guide

12*c* Release 2 (12.1.3)

April 2014

ORACLE®

Preparing to Install the Oracle Tuxedo Application Rehosting Workbench

Installing Oracle Tuxedo Application Rehosting Workbench Using GUI-Mode Installation

Installing Oracle Tuxedo Application Rehosting Workbench Using Silent Installation

Performing Post-Installation Tasks

Oracle Tuxedo Application Rehosting Workbench Supported Platforms

Preparing to Install the Oracle Tuxedo Application Rehosting Workbench

The following sections provide information that you need to know before installing the Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) product software:

- Oracle Universal Installer (OUI)

- Oracle Tuxedo Application Rehosting Workbench Web Distribution

- Hardware and Software Prerequisites

- Oracle Installation Program

- Oracle Home Directory

Oracle Universal Installer (OUI)

The Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) product software installer is based on the Oracle Universal Installer (OUI). To install the Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) product software on your host, you must know how to use OUI to install Oracle products. For more information, see Introduction to Oracle Universal Installer.

OUI is a Java-based installer that enables you to install Oracle components.

Installing Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) with Oracle Home

When you install Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3), an Oracle home is created. Oracle home is managed by OUI.

Oracle Home

An Oracle home is the system context where Oracle products run. It consists of the following:

- Directory location where the products are installed

- Corresponding system path setup

- Program groups associated with the products installed in that home (where applicable)

- Services running from that home

Installing Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3)

To install Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) and create the Oracle home, do the following steps:

1. Run OUI.

2. In the Specify Home Details page, enter the Oracle home settings for the installation session.

3. Continue installation.

Removing Oracle Home

To remove an existing Oracle home, run OUI and click Deinstall Products. In the inventory dialog, select the home(s) you want to delete and click Remove. You can also use the `REMOVE_HOMES` variable in the command line or in a response file.

The syntax is as follows:

```
[On UNIX] ./runInstaller -deinstall -silent
ORACLE_HOME=<LOCATION_OF_ORACLE_HOME>
"REMOVE_HOMES={<LOCATION_OF_ORACLE_HOME_TO_BE_REMOVED>}"
```

Determining the Default Oracle Home

By default, when you start OUI, it searches your system to determine the default Oracle home where Oracle software should be installed.

In each case, the `ORACLE_HOME` name is taken first from the command line if it is specified, or from the response file if specified. If not specified, the following convention is used for the name:

```
Ora<Oracle Home Name>
```

The `ORACLE_HOME` path is taken first from the command line if specified, or from the response file if specified. If not specified, the `ORACLE_HOME` environment variable is used.

If `ORACLE_BASE` is specified in the environment, the `default ORACLE_HOME` starts with `$ORACLE_BASE`.

Multiple Oracle Homes

`OUI` supports the installation of several active Oracle homes on the same host as long as the products support this at run-time. Multiple versions of the same product or different products can run from different Oracle homes concurrently. Products installed in one home do not conflict or interact with products installed on another home. You can update software on any home at any time, assuming all Oracle applications, services, and 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 OUI for installation or deinstallation is the target home. To upgrade or remove products from the target home(s), these products must be shut down or stopped.

Oracle Universal Installer Inventory

The OUI inventory stores information about all Oracle software products installed in all Oracle homes on a host (provided the product installed using Oracle Universal Installer).

Inventory information is stored in Extensible Markup Language (XML) format. The XML format allows for easier problems diagnosis and faster data loading. Secure information is not stored directly in the inventory. As a result, during removal of some products, you may be prompted to enter the required credentials for validation.

Oracle Universal Installer Inventory Structure

The OUI inventory has the following hierarchical structure:

- Central Inventory Pointer File

- Central Inventory

- Oracle Home Inventory

Central Inventory Pointer File

Every Oracle software installation has an associated Central Inventory where details of all Oracle products installed on a host are registered. The Central Inventory is located in the directory that the inventory pointer file specifies. Each Oracle software installation has its own Central Inventory pointer file that is unknown to another Oracle software installation.

For Oracle homes sharing the same Central Inventory, OUI performs all read and write operations on the inventory. The operations on the Central Inventory are performed through a locking mechanism. This implies that when an installation, upgrade, or patching operation occurs on an Oracle home, these operations become blocked on other Oracle homes that share the same Central Inventory.

Below table shows the location of the default inventory pointer file for various platforms:

Linux, LinuxPPC	`/etc/oraInst.loc`
AIX	`/etc/oraInst.loc`
Solaris SPARC, HPUX-IA64	`/var/opt/oracle/oraInst.loc`

In UNIX, if you do not want to use the Central Inventory located in the directory specified by the inventory pointer file, you can use the `-invPtrLoc` flag to specify another inventory pointer file. The syntax is as follows:

```
./runInstaller -silent -invPtrLoc <Location_of_oraInst.loc>
ORACLE_HOME="<Location_of_Oracle_Home>"
ORACLE_HOME_NAME="<Name_of_Oracle_Home>"
```

Note: If the content of the `oraInst.loc` file is empty, OUI prompts you to create a new inventory.

Central Inventory

The Central Inventory contains information relating to all Oracle products installed on a host. It contains the following files and folders:

- Inventory File
- Logs Directory

Inventory File

This file lists all Oracle homes installed on the node. For each Oracle home, it also lists the Oracle home name, home index, and nodes where home is installed. It also mentions if the home is an Oracle Clusterware home or a removed Oracle home. It can only detect removed Oracle homes created using OUI version 11.1 and later.

This file is located at:

```
<central inventory location>/ContentsXML/inventory.xml
```

Note: It is recommended that you do not remove or manually edit this file as it could affect installation and patching.

Logs Directory

The Central Inventory contains installation logs at the following location:

```
<central inventory location>/logs
```

The logs directory contains the logs corresponding to all installations performed on a particular node. You can also find a copy of the installation log in the `$ORACLE_HOME/cfgtoollogs` directory.

The installation logs for an installation are identified by the timestamp associated with the log files. These files are generally saved in the following format:

```
<Name_of_Action><YYYY-MM-DD_HH-MM-SS{AM/PM}>.log
```

For example, consider an attachHome operation performed on 17th, May, 2007 at 6.45AM. The associated log file would be created as follows:

```
AttachHome2007-05-17_06-45-00AM.log
```

Note: The installation logs do not contain any errors or failures.

Oracle Home Inventory

The Oracle home inventory (or local inventory), is present inside each Oracle home. It contains information relevant to a particular Oracle home. This file is located at:

`$ORACLE_HOME/inventory`

It contains the following files and folders:

- Components File

  This file contains details about third-party applications (like Java Runtime Environment (JRE)), required by different Java-based Oracle tools and components. In addition, it also contains details of all the components as well as patchsets or interim patches installed in the Oracle home. This file is located at:

  `ORACLE_HOME/inventory/ContentsXML/comps.xml`

- Home Properties File

  This file contains node list details, the local node name, and the CRS flag for the Oracle home. In a shared Oracle home, the local node information is not presented. This file also contains the following information:

  - GUID — Unique global ID for the Oracle home

  - ARU ID — Unique platform ID. The patching and patchset application depends on this ID.

  - ARU ID DESCRIPTION — Platform description

  The information in `oraclehomeproperties.xml` overrides the information in `inventory.xml`. This file is located at:

  `$ORACLE_HOME/inventory/ContentsXML/oraclehomeproperties.xml`

- Other Folders

  The following table lists the other folders you can find in the Oracle home inventory:

Folder Name	Description
Scripts	Contains the scripts used for the cloning operation.
ContentsXML	Contains the details of the components and libraries installed.

Templates	Contains the template files used for cloning.
oneoffs	Contains the details of the one-off patches applied.

Creating the Central Inventory

OUI enables you to set up the Central Inventory on a clean host or register an existing Oracle home with the Central Inventory when it is lost or corrupted. If the Central Inventory does not exist, OUI creates the Central Inventory in the location specified by the oraInst.loc file.

You can set up the Central Inventory by using the -attachHome flag of Oracle Universal Installer. The syntax is as follows:

```
./runInstaller -silent -attachHome -invPtrLoc ./oraInst.loc

ORACLE_HOME="<Oracle_Home_Location>" ORACLE_HOME_NAME="<Oracle_Home_Name>"

"CLUSTER_NODES={<node1,node2>}" LOCAL_NODE="<node_name>"
```

Using the Session Variables

You can use the following session variables:

- ORACLE_HOME
- ORACLE_HOME_NAME

Verifying the Operation

After attaching the Oracle home, you can verify the success of the operation by verifying the contents of the log file located in the <central_inventory>/logs directory. You can also view the contents of the inventory.xml file under the <central-inventory>/ContentsXML directory to verify if the Oracle home is registered.

Detaching Oracle Homes from the Central Inventory

You can detach an Oracle home from the Central Inventory. When you pass the flag, it updates the inventory.xml file presented in the Central Inventory. The syntax is as follows:

```
./runInstaller -silent -detachHome -invPtrLoc ./oraInst.loc

ORACLE_HOME="<Oracle_Home_Location>" ORACLE_HOME_NAME="<Oracle_Home_Name>"
```

Using Optional Flags

If you are using a shared Oracle home, use the `-cfs` flag. This ensures that the local node information is not populated inside a shared Oracle home.

```
./runInstaller -silent -cfs -detachHome -invPtrLoc ./oraInst.loc

ORACLE_HOME="<Oracle_Home_Location>" ORACLE_HOME_NAME="<Oracle_Home_Name>"

./runInstaller -silent -detachHome -invPtrLoc ./oraInst.loc

ORACLE_HOME="<Oracle_Home_Location>" ORACLE_HOME_NAME="<Oracle_Home_Name>"
```

You can completely clean the Oracle home and remove the home directory using the -removeallfiles flag. The syntax is as follows:

```
./runInstaller -silent -deinstall -removeallfiles -invPtrLoc ./oraInst.loc

ORACLE_HOME="<Oracle_Home_Location>" ORACLE_HOME_NAME="<Oracle_Home_Name>"
```

If you want to disable the warning message that appears when you use the `-removeallfiles` flag, use the `-nowarningonremovefiles` flag. The syntax is as follows:

```
./runInstaller -silent -deinstall -nowarningonremovefiles -invPtrLoc
./oraInst.loc

ORACLE_HOME="<Oracle_Home_Location>" ORACLE_HOME_NAME="<Oracle_Home_Name>"
```

Removing the Central Inventory

Even after all Oracle homes on a host are removed, you will find traces of the inventory with certain log files. If you do not want to maintain these files and want to remove the Central Inventory, do the following:

Removing the Central Inventory on UNIX Platforms

You can remove the Central Inventory on UNIX. Do the following steps:

1. Locate the `oraInst.loc` file and get the Central Inventory location (`inventory_loc` parameter) from this file.

    – For Solaris, it is located in the `/va/opt/oracle` folder.

    – For Linux, it is located in the `/etc` folder.

2. Remove the Central Inventory by executing the following command:

    ```
    rm -rf <central_inventory_location>
    ```

3. Remove the `oraInst.loc` file by executing the following command with root privileges:

 – Solaris: `rm /va/opt/oracle/oraInst.loc`

 – Linux: `rm /etc/oraInst.loc`

Backing up the Inventory

You can back up the Oracle home using your preferred method. You can use any method (such as zip, tar, and cpio), to compress the Oracle home.

It is highly recommended to back up the Oracle home before any upgrade or patch operation. You should also back up the Central Inventory when Oracle home is installed or deinstalled.

Recovering from Inventory Corruption

The inventory (Central and the Oracle home inventory) is critically important in Oracle software life-cycle management. The following sections explain what you need to do in case of inventory corruption.

Customizing and Creating Response Files

This section introduces you OUI response file availability for silent and suppressed installations. This section also describes how to modify or create a response file so you can customize and standardize Oracle products installation in your organization.

About Response Files

What Is a Silent Installation?

A silent installation runs in the console and does not use the GUI. The interactive dialogs you normally see are not displayed during silent installation. Instead of prompting you to select a series of installation options, OUI installs the software using a predefined set of options stored in the response file or entered in the command line. You can view silent installation progress in the console.

What Is a Response File?

A response file is a specification file containing information you normally enter through the OUI user interface during 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 installation type can be set automatically within the response file.

Why Perform a Silent Installation?

Silent installation 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 silent installation, you should review the settings in the response file template provided with your Oracle product.

Notes:

- If you attempt to perform silent installation on a UNIX computer where no Oracle products have been installed, OUI uses the default inventory location, then prompts you to run the `oraInstRoot.sh` script with root privileges upon successful installation.

- The script is saved in the `_*Central Inventory - oraInventory *_` directory. This script sets up the Central Inventory on a clean host. You can override the default location by setting it in `INVENTORY_LOCATION`.

- This location is ignored if a Central Inventory already exists and is pointed to `/var/opt/oracle/oraInst.loc` (or `/etc/oraInst.loc` for Linux, AIX, and Sequent).

    For more information, see "Oracle Universal Installer Inventory" on page 1-3.

Creating a Response File With Record Mode

You can create a new response file, based on the installation options you select, using OUI record mode.

When you use record mode, OUI records the installation session into a response file. You specify the name of the response file in the command line. The recorded response file is generated immediately after the Summary page, so you do not need to actually install your Oracle product to create the response file. That is, 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 record mode on UNIX systems.

Using Record Mode

To record a new response file:

1. At the command prompt, use the cd command to change to the directory that contains the Oracle Universal Installer executable file (runInstaller.sh) for your installation.

2. Enter the following command:

```
./runInstaller -record -destinationFile <response_file_name> (on UNIX)

Replace the <response_file_name>
```

with the complete path for the new response file. For example:

On UNIX:

```
./runInstaller -record -destinationFile
/private/temp/install_oracle11g.rsp
```

3. Use the OUI user interface to select your installation options. These will be recorded.

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

OUI saves your new response file using the path and file name you specified on the command line.

Response File Format

The following sections describe the organization and content of an OUI response file.

Note: It is recommended that you use an absolute path for the file name. However, if the file name is a relative path, it should be relative to the directory where the oraparam.ini file is presented.

Variable Values

Variable Lookup Order

All variable values within a response file are in the name-value format. If two components have a variable with the same name, the expression should be written as follows to preserve the uniqueness of each variable:

```
<component>:<variable>=<value>
```

Component Nomenclature and Version Ambiguity

If two components have the same internal name, but are different versions, resolve the ambiguity by specifying the expression as follows:

```
<component>:<version>:<variable>=<value>
```

Order of Variables

OUI looks for command line variables in the following order:

```
<component>:<version>:<variable>=<value>
```

```
<component>:<variable>=<value>
```

```
<variable>=<value>
```

Similarly, if command line variables are not found, OUI looks for variables in the response file in the same order as shown above.

Note: OUI treats incorrect context, format, or type values within a response file as if no value were specified.

Response File Entries Order

There are no restrictions on where you place entries in the response file. You can insert entries in any order desired. You can also subsequently move existing entries to any other position within the file. Be aware, however, that although the order of the entries is unimportant, naming conventions require that the variable names must be unique regardless of where they appear in the file.

Setting the Recommendation Value

Values for variables are specified as:

```
<variable> = <recommendation> : <value>
```

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

For values given as `<value_unspecified>`, you can optionally specify a value, where `<value>` can be one of the following types listed in below table.

Type	Representation
Number	10
Boolean	TRUE or FALSE (case insensitive)
String	"Value"
String List	{"value1", "value2"}

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

- If you specify Forced, no dialog appears during installation. The value is automatically used. You cannot change the value.

- If you specify Default, the dialog appears during installation with the value as the default. You can choose another value if desired.

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

Comments

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

The header is a comment that 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 on UNIX platforms.

#Example : UNIX_GROUP_NAME = "install"

UNIX_GROUP_NAME=<Value Unspecified>
```

Response File Parameters

The following list describes OUI parameters. All products installed using OUI have these parameters in addition to product-specific variables.

Note: The parameters provided below are OUI-specific and not necessarily the parameters applicable for your platform and version. Always check your specific installation guide for the list of valid response file parameters.

For Oracle Tuxedo Application Rehosting Workbench specific response file parameters, see "Required Variables in Oracle Tuxedo Application Rehosting Workbench Response File" on page 4-2.

INCLUDE

Specifies the list of response files to be included in a particular response file. If you want to include other response files in your main response file, you must use this parameter. Note that the values in the main response file have priority over the included response files.

```
INCLUDE={"file1.rsp","file2.rsp",............."filen.rsp"}
```

You should specify the absolute path in the INCLUDE statement for each response file to be included. If you wish to specify a relative path, note that the location is relative to the current working directory (the location of the oraparam.ini file).

RESPONSEFILE_VERSION

Specifies the version number of the response file.

```
RESPONSEFILE_VERSION = <version_number>
```

FROM_LOCATION

The location of the source of the products to be installed. The default generated value for this parameter is a path relative to the location of the runInstaller file oraparam.ini. Relative paths are necessary for shared response files used by multiple users over a network. Since people may be mapping to different drives, absolute paths will not work for shared response files.

Notes:

- You must enter a value for FROM_LOCATION for a complete silent installation. You may want to use the command line to set this parameter. If the location is a relative path, remember that the path should be relative to the location of the `oraparam.ini` file.

- Make sure that `/var/opt/oracle/oraInst.loc` (or `/etc/oraInst.loc` for Linux, AIX, and Sequent) exists. The response file for UNIX has a public variable, `setunixinstallgroup`, that only becomes effective when it is the first OUI installation and the `oraInst.loc` file is not already present at the location mentioned above.

FROM_LOCATION_CD_LABEL

Used in multiple-CD installations. It includes the label of the compact disk where the file `products.jar` exists. You can find the label in the `disk.label` file in the same directory as `products.jar`.

LOCATION_FOR_DISK2

This parameter is for the complete path to other disks.

```
LOCATION_FOR_DISK2="F:\teststage\cd\Disk2"
```

If there are more than two disks, more variables will be added as LOCATION_FOR_DISK3, and so on.

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

- OUI looks for the `Location_For_Disk{DiskNumber}` variable in the response file and uses that location.

- If the variable does not have a value or does not have the required files, it looks for the components under `../../Disk{DiskNumber}/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:

```
<Product_Shiphome_Location>/stage
```

RESTART_SYSTEM

Set this boolean variable to TRUE if you want to restart the system without user confirmation. This is the force value for restarting the system.

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.

NEXT_SESSION = TRUE

NEXT_SESSION_ON_FAIL

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

NEXT_SESSION_ON_FAIL = TRUE

NEXT_SESSION_RESPONSE

Use this string variable to enter the complete path of the response file for the next session 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, OUI ignores the value of this variable.

NEXT_SESSION_RESPONSE="/private/usr2/nextinstall.rsp"

ORACLE_HOME

The location where products are to be installed. You must enter a value for ORACLE_HOME for a complete silent installation.

ORACLE_HOME_NAME

The name of the current Oracle home. You must enter a value for ORACLE_HOME_NAME for a complete silent installation.

SHOW_COMPONENT_LOCATIONS_PAGE

The location page, which appears in a custom installation type, can be suppressed by setting this value to FALSE. If you set the value to FALSE, you are 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 deselected. This page appears only during custom installation type.

SHOW_END_SESSION_PAGE

The installation 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 you need to show the confirmation when exiting the installer.

```
SHOW_EXIT_CONFIRMATION = TRUE
```

SHOW_INSTALL_PROGRESS_PAGE

The installation progress page, which appears during the installation phase, can be suppressed by setting the value to FALSE.

SHOW_OPTIONAL_CONFIG_TOOL_PAGE

Set to TRUE if you must show the Optional Configuration Tools page in the installer. The Optional Configuration Tools page shows a 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_ROOTSH_CONFIRMATION

Set to TRUE if you need to show the Confirmation dialog asking to run the root.sh script in the installer. This variable is valid only on UNIX platforms.

```
SHOW_ROOTSH_CONFIRMATION = TRUE
```

SHOW_SPLASH_SCREEN

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

```
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 you need to show the Welcome page on the installer.

`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 a string list. 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, the second representing the version.

For example, RDBMS 11.1. may be represented as `{"oracle.rdbms","11.1.0.0.0"}`.

UNIX_GROUP_NAME

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

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

REMOVE_HOMES

Use to identify the Oracle home(s) you want to remove from the inventory during a deinstallation session. For each home, specify the home name using the full path information.

`REMOVE_HOMES={"/home/oracle/ora9i","/home/oracle/ora8i"}`

DEINSTALL_LIST

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

`DEINSTALL_LIST={"sample1","1.0.0.0.0"}`

SHOW_DEINSTALL_CONFIRMATION

Set to `FALSE` if you want to hide the deinstall confirmation dialog box during silent deinstallation.

SHOW_DEINSTALL_PROGRESS

Set this parameter to `FALSE` if you want to hide the deinstallation progress dialog box during silent deinstallation.

DEPENDENCY_LIST

List of dependents on which the component depends. These dependent components represent the list of components that appear as "selected" during installation. Following is a list of a few parameter characteristics:

- 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 deselected items in a dialog.

- Specifying required dependents is redundant since they will be selected anyway. The dependents selection can have two other types of dependents: optional and one-or-more.

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

A typical example is listed here:

```
DEPENDENCY_LIST={"oracle.netclt","9.2.0.4.0","oracle.netmgr","9.2.0.4.0"}
```

CLUSTER_NODES

Lists the nodes on the cluster where Oracle home is installed or would be installed. For new installations, this would be the node list where the installation needs to be done (including the local node). For patching or upgrades, this is the node list where Oracle home is already installed.

```
"CLUSTER_NODES={alpha-1, alpha-2}"
```

REMOTE_NODES

Lists the remote nodes (apart from the local node), on the install cluster during a silent installation. OUI installs on all named nodes.

```
"REMOTE_NODES={alpha-1, alpha-2}"
```

LOCAL_NODE

This parameter specifies the current node where the installation occurs.

```
LOCAL_NODE={}
```

OPTIONAL_CONFIG_TOOLS

While all the required configuration tools are launched by the installer, you can control the optional configuration tools you would want to launch by specifying the tool 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 also call the Net Configuration Assistant or the Database Configuration Assistant at the end of a database installation in silent mode. For example, to launch the Net Configuration Assistant in silent mode, you can pass the parameter s_responseFileName="netca.rsp".

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

For Example:

OPTIONAL_CONFIG_TOOLS = {"configtool2 ", "configtool3"}

OPTIONAL_CONFIG_TOOLS=<Value Unspecified>

If no value is specified for this variable, all the tools are run by default. If there is a value specified, only those optional tools are run while the rest of the tools are ignored.

Suppress the configuration tool by mentioning only the tools that you want to run as part of the OPTIONAL_CONFIG_TOOLS variable added for each component. You should use the configuration tool internal names.

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

For example, if oracle.server has Tool1 and Tool2 and you want to run only Tool1 in the response file, you can specify it as follows:

oracle.server:11.1.0.0.0:OPTIONAL_CONFIG_TOOLS={"Tool1"}

INSTALL_TYPE

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

The installation type variable is only generated for top-level components and only when there is more than one installation type available.

Note: You must enter a value for INSTALL_TYPE.

SELECTED_LANGUAGES

You can set the languages for component installation. You must use the internal name while specifying the value:

```
en, : English

ja, : Japanese
```

For example, to specify Japanese:

```
SELECTED_LANGUAGES = {"ja"}
```

Note: The SELECTED_LANGUAGES variable is only generated when more than one language is available.

Installing with a Response File

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

These are called silent installations. Instead of prompting you to select a series of installation options, OUI installs the software using a predefined set of options. These options are stored in a response file (.rsp).

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 on such a computer, you must first run the oraInstRoot.sh script, which is saved in the /oraInventory directory. You must run this script with root privileges. This enables OUI to set up the Central Inventory on a clean host.

For more information, see "Oracle Universal Installer Inventory" on page 1-3.

Specifying a Response File

To start OUI and specify the response file, enter the following on the command line in the directory where the executable file is installed:

On UNIX: ./runInstaller -responseFile <filename> <optional_parameters>

Note: You must specify the complete responseFile path. If you do not, OUI assumes the location to be relative to the oraparam.ini file associated with the launched OUI.

For help on command line usage, enter the following on the command line in the directory where the executable file is stored:

On UNIX:

```
./runInstaller -help
```

Optional Parameters When Specifying a Response File

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

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

- `-silent` — Used with the `-responseFile` flag to run OUI in complete silent mode. Note that the Welcome dialog is suppressed automatically.

- In a file named `silentInstall<timestamp>.log` for hosts without an Oracle inventory. This file is generated in the `/tmp` directory (UNIX).

- In the inventory logs directory for hosts that already had an inventory.

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

Setting Response File Variables From the Command Line

With OUI 2.1 and higher, you can specify the value of certain variables when you start OUI from the command line. Specifically, you can specify session and component variables.

For more information, see "Response File Format" on page 1-11.

Note: 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, use the following command syntax:

On UNIX:

```
./runInstaller session:<variable_name>=<value>
```

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

On UNIX:

```
./runInstaller session:SHOW_WELCOME_PAGE=false
```

Note: The "`session:`" tag is optional and is used mainly to remove any possible ambiguity.

Specifying the Value of a Component Variable

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

On UNIX:

```
./runInstaller
<component_name>:<component_version>:<variable_name>=<value>
```

Modes of Installation

You can use OUI to install Oracle products in any of the three following modes:

- Interactive:

  Use OUI interactive mode to use the graphical user interface to walk through the installation, providing information in the installation dialogs when prompted. This method is most useful when installing a small number of products in different setups on a small number of hosts.

- Suppressed:

  Use OUI suppressed mode to supply the necessary information by using a combination of a response file or command line entries with certain interactive dialogs. You can choose which dialogs to suppress by supplying the information on the command line when you invoke OUI. This method is most useful when an installation has a common set of parameters that can be captured in a response file, in addition to custom information that must be input by hand.

- Silent:

  Use OUI silent installation mode to bypass the graphical user interface and supply the necessary information in a response file. This method is most useful when installing the same product multiple times on multiple hosts. By using a response file, you can automate the installation of a product for which you know the installation parameters.

Installation Media

Note: When you invoke `runInstaller` (UNIX), you should invoke it from the directory where this command is present, or you must specify the complete path to `runInstaller` (UNIX).

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 receive an `Xlib` error or a `"Failed to connect to Server"` error when you are running OUI on the Solaris operating system, do the following:

1. Define the following environment variable on the host computer where you are running OUI:

`%setenv DISPLAY <machine name>:0.0`

2. Replace `<machine name>` with the name of the computer that will display OUI.

3. On the computer that will display OUI, enter the following command, which allows other computers to display information on the computer monitor: `%xhost +`

4. Rerun the `runInstaller` script after you have set the `DISPLAY` environment variable.

Note: You can run OUI without specifying the `DISPLAY` variable by running in silent mode using a response file.

Providing a UNIX Installer Location with Root Privileges

You must have root privileges to perform various UNIX installation operations. For example, you must have root privileges to be able to create the OUI inventory.

If you are installing OUI for the first time, you are prompted to run a shell script from another terminal window before proceeding with the installation. OUI prompts you to run `root.sh` after installation completes only if the script is required to run as root before configuration assistants are run. Otherwise, you are prompted to run `root.sh` as root later.

Note: When running OUI in silent mode, if `root.sh` is required prior to configuration assistants, OUI skips configuration assistants during the installation. You must run `root.sh` as root and then run the skipped configuration assistants after the silent installation is complete.

To successfully run the required shell script:

1. Leave the OUI 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 `./root.sh`.

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

Note: Do not exit the installation to run the shell script. Exiting the installation 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 also prompts you to provide the name of the group that owns the base directory.

You must choose a UNIX group name that has permissions to update, install, and remove Oracle software. Members of this group must have write permissions for the chosen base directory.

Only users who belong to this group are able to install or remove software on this host.

Deinstalling Oracle Products

Removing Oracle Products and Oracle Homes

You can deinstall Oracle products before selecting products to install, or after a successful installation.

To remove an Oracle product or Oracle home using interactive mode, do the following steps:

1. Start OUI from a CD-ROM or:

   – For UNIX platforms, from the command line, run the script called `runInstaller` from the directory where it is stored by default at the same level as the first Oracle home created on that host.

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

   The Inventory panel appears.

3. Select the product(s) you want to remove from the **Contents** tab of the Inventory panel and click **Remove**. You can also remove Oracle homes in the same manner. After you have removed an Oracle home, you can reuse its name and location to install other products.

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

   OUI warns you of any product dependencies that might cause problems if particular products are removed, and prompts you to confirm the deinstallation.

   Pay special attention to the full list of products being removed before proceeding. OUI computes this list based on the dependencies of each component.

Note: You can also remove products by using the OUI **Installed Products** button as long as you perform this action before selecting products to install.

Deinstalling Top-level Products With Dependents

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

When you select a specific component for removal, OUI analyzes the dependency information to determine if other components should be removed along with it. Generally, if a component is selected for removal, the following components are removed with it:

All components that have a required dependency on the selected component.

Dependents of the selected component that have no other dependents. A dependent is a component on which the top-level component (dependent), has a dependency.

Silent Deinstallation

Not only can you perform command line installations, you can also perform command line deinstallations. A command line deinstallation enables you to remove Oracle products or Oracle homes from your system without using the OUI graphical user interface.

You can choose to display no dialog boxes or prompts, 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 commands to immediately display the Inventory dialog box, which allows you to select items for removal without navigating the OUI startup screen:

`./runInstaller -deinstall -silent` (on UNIX)

Hiding the Inventory Dialog Box

If you want to hide the inventory dialog box during a deinstallation, you can specify the products to be removed in the response file `DEINSTALL_LIST` parameter; specify Oracle homes to be removed with the `REMOVE_HOMES` variable.

As with other response file parameters, you can also specify the `DEINSTALL_LIST` parameter on the OUI command line. For example, on a UNIX machine, enter:

```
./runInstaller -deinstall -silent
DEINSTALL_LIST={"component1","1.0.1.2.0"}
```

To remove Oracle homes from the inventory, use the `REMOVE_HOMES` variable.

Hiding the Deinstallation Confirmation and Progress Dialog Boxes

Use the following commands to hide the deinstallation confirmation and progress dialog boxes during a command line deinstallation:

On a UNIX system:

```
./runInstaller -deinstall -silent session:SHOW_DEINSTALL_PROGRESS=false
session:SHOW_DESINSTALL_CONFIRMATION=false
```

About Oracle Universal Installer Log Files

When you install or deinstall products using OUI, important information about each installation is saved not only in the inventory, but also in a series of log files, located in the following directory:

```
$ORACLE_HOME/cfgtoollogs
```

You can use these log files to troubleshoot installation problems. These files are also crucial for removing and configuring the various software components you install on your UNIX computer. OUI displays the name and location of the current session log file on the Install page. Each installation or configuration utility provides a separate folder containing the logs inside the `$ORACLE_HOME/cfgtoollogs` folder.

Note: The logs used to remove products are different from the `installActions<timestamp>.log` generated during the install process. The `installActions<timestamp>.log` is easier to read and can be used to view the operations performed at installation time.

Oracle Internationalization

Installation Dialogs Language

OUI runs in the operating system language. OUI uses the language that Java detects, the system locale value, and sets that to the default language. OUI dialogs are displayed in this language if available. If specific OUI dialogs are not translated in the language of the operating system, these dialogs are shown in English.

OUI displays the translated GUI only if the variable `NLS_ENABLED` has been set to `TRUE` in the `oraparam.ini` file. If the `NLS_ENABLED` variable is set to `FALSE`, all text is shown in English.

Note: The dialogs displayed for internationalization can only be customized parts; some of them are embedded in OUI.

Oracle Tuxedo Application Rehosting Workbench Web Distribution

An evaluation copy of Oracle Tuxedo Application Rehosting Workbench is available for download from the Oracle corporate Web site at http://www.oracle.com/technology/software/index.html.

Platform-specific installer files for the Oracle Tuxedo Application Rehosting Workbench product software are available for download from the Oracle corporate Web site.

Hardware and Software Prerequisites

System Requirements

The system requirements for Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) are given in Table 1-1.

Table 1-1 The System Requirements for Oracle Tuxedo 12*c* Release 2 (12.1.3)

Component	Requirement
Platform*	Any platform identified in "Oracle Tuxedo Application Rehosting Workbench Supported Platforms" on page A-1.
Hard disk drive	As stated in the data sheet for the target platform in "Oracle Tuxedo Application Rehosting Workbench Supported Platforms" on page A-1.
Memory	As stated in the data sheet for the target platform in "Oracle Tuxedo Application Rehosting Workbench Supported Platforms" on page A-1.

Temporary Storage Space Requirements

The Oracle Installation program uses a temporary directory in which it extracts the files from the archive that are needed to install Oracle Tuxedo Application Rehosting Workbench on the target system. During the installation process, your temporary directory must contain sufficient space to accommodate the compressed Java Runtime Environment (JRE) bundled with the installer and an uncompressed copy of the JRE that is expanded into the temporary directory. The installation program moves the JRE from the temporary directory to the *Oracle Home directory* at the end of the installation process. For information about the Oracle Home directory, see "Oracle Home Directory" on page 1-31.

The amount of temporary storage space needed depends upon the target platform, as stated in the data sheets in "Oracle Tuxedo Application Rehosting Workbench Supported Platforms" on page A-1.

When you start OUI, it automatically copies some executable files and link files into the default `/tmp` directory on the machine. If the machine is set to run cron jobs periodically (along with many other processes that may be running), these jobs attempt to clean up the default temporary directory, thereby deleting some files and causing OUI to fail.

To ensure there is adequate temporary space, you may want to allocate an alternate directory for use as a temporary directory for the installation. If there are any cron jobs or processes that are automatically run on the machines to clean up the temporary directories, ensure you set the TMP or TEMP environment variable to a different location (other than the default location) that is secure on the hard drive (meaning a location on which the cleanup jobs are not run). Also ensure that you have write permissions on this alternative TEMP directory. This must be done before you execute `runInstaller`.

Note: Specifying an alternative temporary directory location is not mandatory, and is required only if any cron jobs are set on the computers to clean up the `/tmp` directory.

Oracle Installation Program

The Oracle Tuxedo Application Rehosting Workbench software is distributed as an installer file, which also contains a copy of the Oracle Installation program. The Oracle Installation program is the Oracle standard tool for installing the Oracle Tuxedo software on UNIX systems.

Installation Methods

The Oracle Installation program supports two installation methods. You can use any of these methods to install the Oracle Tuxedo Application Rehosting Workbench product software:

- Graphical user interface (GUI) installation, described in "Installing Oracle Tuxedo Application Rehosting Workbench Using GUI-Mode Installation" on page 3-1.

- Silent installation, described in "Installing Oracle Tuxedo Application Rehosting Workbench Using Silent Installation" on page 4-1.

Cancelling Installation

Clicking the "Cancel" or the "close' window button in GUI mode creates an incomplete Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) installation. You will have to re-install Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3).

If you installed Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) in a previous Oracle Tuxedo Application Rehosting Workbench product directory, and you want to return to your original configuration, you must re-install your previous Oracle Tuxedo Application Rehosting Workbench version.

Oracle Home Directory

When you install Oracle Tuxedo Application Rehosting Workbench, you are prompted to specify an Oracle Home directory. The Oracle Home directory is a repository for common files that are used by multiple Oracle products installed on the same machine.

An Oracle home is the system context where Oracle products run. This context consists of the following:

- Directory location where the products are installed

- Corresponding system path setup

- Program groups associated with the products installed in that home (where applicable)

- Services running from that home

Choosing an Oracle Home Directory

During the installation of Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3), you are prompted to input a new or choose an existing Oracle Home directory.

Installation Road Map

You are now ready to begin your installation. To install Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3), see one of the following sections:

- "Installing Oracle Tuxedo Application Rehosting Workbench Using GUI-Mode Installation" on page 3-1

- "Installing Oracle Tuxedo Application Rehosting Workbench Using Silent Installation" on page 4-1

If you want to uninstall your Oracle Tuxedo Application Rehosting Workbench software, see "Performing Post-Installation Tasks" on page 4-1.

Installing Oracle Tuxedo Application Rehosting Workbench Using GUI-Mode Installation

The following sections describe how to install Oracle Tuxedo Application Rehosting Workbench using graphical user interface (GUI) mode installation on UNIX systems:

- What Is GUI-Mode Installation?
- Starting GUI-Mode Installation on a UNIX System
- Running GUI-Mode Installation

What Is GUI-Mode Installation?

The graphical user interface mode installation is the graphics-based method of executing the Oracle Installation program. It can be run on UNIX system.

To run GUI-mode installation, the console attached to the machine on which you are installing the software must support a Java-based GUI. Not all consoles for UNIX systems support Java-based GUIs.

Note: To install Oracle Tuxedo Application Rehosting Workbench on a UNIX system with a non-graphics console, use silent install mode.

Starting GUI-Mode Installation on a UNIX System

To start the GUI-mode installation process on a UNIX system, follow these steps:

1. Select a UNIX system that meets the hardware and software requirements described in "Oracle Tuxedo Application Rehosting Workbench Supported Platforms" on page A-1

2. Log in to the UNIX system as the Oracle Tuxedo Application Rehosting Workbench administrator.

3. Ensure that you have enough free space for the Oracle Tuxedo Application Rehosting Workbench installation.

   For disk space requirements, see "Oracle Tuxedo Application Rehosting Workbench Supported Platforms" on page A-1.

4. Install Oracle Tuxedo Application Rehosting Workbench by downloading from the Oracle Web site:

   a. Go to http://www.oracle.com/technology/software/index.html and download the Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) installation file specific to your platform.

   b. Go to the directory where you downloaded the installer, unzip the installer file, and then go to the `Disk1\install` directory, run the `runInstaller` to invoke the installation procedure.

   Note: GUI mode is the default for Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) installation. If a GUI interface is not available on your Unix platform, you will receive an error message and the installation process is aborted.

   If a GUI interface is not available on your Unix platform, you must use silent mode installation. For more information, see What Is Silent Installation?.

5. Proceed to "Running GUI-Mode Installation" on page 3-2.

Running GUI-Mode Installation

The Oracle Tuxedo Application Rehosting Workbench installer program prompts you to enter specific information about your system and configuration. For instructions on responding to the prompts during installation, see Table 3-1.

Table 3-1 Instructions

In This Window . . .	Perform the Following Action . . .
Welcome	Click **Next** to proceed with the installation. You may cancel the installation at any time by clicking Cancel.
Specify Inventory directory (Unix only)	On UNIX operating systems, this screen appears if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location.
Specify Oracle Home	Name: Enter a name for the Oracle home. This name identifies the program group associated with a particular home and the Oracle services installed on this home. The Oracle home name must be between 1 to 127 characters long, and can include only alphanumeric characters and underscores.
	Enter the full path to an Oracle home, or select an Oracle home from the drop-down list of existing Oracle homes. The Oracle home location is the directory where products are installed. Data files may or may not be installed within an Oracle home. You can use the Browse button to choose a directory to install your product. Different homes cannot share the same location.
	Oracle Tuxedo Application Rehosting Workbench Installation Directory is "`$ORACLE_HOME/artwb12.1.3.0.0`" by default (UNIX) and cannot changed after the `ORACLE_HOME` is determined. The installer doesn't allow to install two same version tuxedo product into the same `ORACLE_HOME`.
	If the selected `ORACLE_HOME` already installed the same version Oracle Tuxedo Application Rehosting Workbench, the installer will show a warning.
Install	Select (add) or deselect (clear) one or more software components from the selected install type, or choose one of the other four install types or Custom Set from the drop-down list menu and customize its software components.
	After selecting or deselecting one or more software components from the selected install set, click Next to continue with the installation.

Table 3-1 Instructions

In This Window . . .	Perform the Following Action . . .
Samples install Confirm	It will prompt customers to choose whether install the sample files.
Summary	Click Install to start the installation.
Installing	Nothing to do.
Install End	Click Exit to exit the installation program.

Congratulations! Your installation of the Oracle Tuxedo Application Rehosting Workbench software is complete!

Installing Oracle Tuxedo Application Rehosting Workbench Using Silent Installation

The following sections describe how to install Oracle Tuxedo Application Rehosting Workbench using silent installation on UNIX systems:

- What Is Silent Installation?
- Using Silent Installation: Main Steps

What Is Silent Installation?

Silent installation reads the settings for your configuration from a text file that you create prior to beginning the installation. Manual intervention is not necessary during the installation process. Silent installation works on UNIX systems.

Silent installation is a way of setting installation configurations only once and then using those configurations to duplicate the installation on many machines.

Using Silent Installation: Main Steps

Creating a Response File

You can create a new response file, based on the installation options you select, by using the OUI record mode.

When you use record mode, OUI records the installation session to 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; you do not need to actually install your Oracle product to

create the response file. That is, 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 are 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 this case, the variable values specified in the original source response file will be recorded to the new response file.

The following sections describe how to use record mode on UNIX systems.

Using Record Mode

To record a new response file:

1. At the command prompt, use the `cd` command to change to the directory that contains the OUI executable file (`runInstaller.sh`) for your installation.

2. Enter the following command on UNIX:

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

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

   On UNIX:

   ```
   ./runInstaller -record -destinationFile
   /private/temp/install_oracle11g.rsp
   ```

3. Use the OUI user interface to select and record your installation options.

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

OUI saves your new response file using the path and file name you specified on the command line.

Required Variables in Oracle Tuxedo Application Rehosting Workbench Response File

The variables defined in this section are specific for Tuxedo installer.

Note: For String type variables, you must contain the value using the double quote.

ORACLE_HOME

The location where products are to be installed. You must enter a value for ORACLE_HOME for a complete silent installation.

ORACLE_HOME_NAME

The name of the current Oracle home. You must enter a value for ORACLE_HOME_NAME for a complete silent installation.

SAMPLES_INSTALL

If install samples for Oracle Tuxedo Application Rehosting Workbench products set it to true. It is required.

OVER_WRITE

Overwrite the previous installation if the value is true. It is required if Oracle Tuxedo Application Rehosting Workbench product with the same version was already install on this ORACLE_HOME and you want to overwrite it.

Installing with a Response File

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

This is called silent installation.

Instead of prompting you to select a series of installation options, OUI installs the software using a predefined set of options. These options are stored in a response file (.rsp).

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 in this situation, you must first run the oraInstRoot.sh script located in the /oraInventory directory. You must run this script with root privileges. This enables OUI to set up the Central Inventory on a clean host.

For more information, see "Oracle Universal Installer Inventory" on page 1-3.

Specifying a Response File

To start OUI and specify the response file, enter the following command on the command line in the directory where the executable file is installed:

On UNIX:

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

Notes:

- To install Oracle Tuxedo Application Rehosting Workbench in complete silent mode, you must specify `<optional_parameters>` to `-silent`.

- You must specify the complete `responseFile` path. If you do not, OUI assumes the location is relative to the `oraparam.ini` file associated with the launched OUI.

For help on command line usage, enter the following on the command line in the directory where the executable file is stored:

On UNIX:

```
./runInstaller -help
```

Optional Parameters When Specifying a Response File

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

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

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

- `-waitforcompletion` - Use the `-waitforcompletion` flag with `-silent` flag to wait for completion instead of spawning the java engine and exiting.

- In a file named `silentInstall<timestamp>.log` for hosts without an Oracle inventory. This file is generated in the `/tmp` directory on UNIX platforms.

- In the inventory logs directory for hosts that already had an inventory.

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

UNIX Template Sample File

This sample UNIX response file in Listing 4-1 applies to a silent installation of Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3).

Listing 4-1   UNIX response File

```
######################################################################
## Copyright (c) 1999, 2013 Oracle. 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"}      ##
##                                                             ##
## The values that are given as <Value Required> need to be    ##
## specified for a silent installation to be successful.       ##
##                                                             ##
##                                                             ##
## This response file is generated by Oracle Software          ##
## Packager.                                                   ##
######################################################################
```

```
RESPONSEFILE_VERSION=2.2.1.0.0


#--------------------------------------------------------------------------
------
#Name        : UNIX_GROUP_NAME
#Datatype   : String
#Description: Unix group to be set for the inventory directory. Valid only
in Unix platforms.
#Example: UNIX_GROUP_NAME = "install"
#--------------------------------------------------------------------------
------
UNIX_GROUP_NAME="oinstall"


#--------------------------------------------------------------------------
------
#Name        : FROM_LOCATION
#Datatype   : String
#Description: Complete path to the products.xml.
#Example: FROM_LOCATION = "../stage/products.xml"
#--------------------------------------------------------------------------
------
FROM_LOCATION="/nfs/users/huchchen/art/stage_art_linux/Disk1/stage/product
s.xml"


#--------------------------------------------------------------------------
------
#Name        : FROM_LOCATION_CD_LABEL
#Datatype   : String
#Description: This variable should only be used in multi-CD installations.
It includes the label of the compact disk where the file "products.xml"
```

exists. The label can be found in the file "disk.label" in the same directory as products.xml.

```
#Example: FROM_LOCATION_CD_LABEL = "CD Label"

#-----------------------------------------------------------------------------
------

FROM_LOCATION_CD_LABEL=<Value Unspecified>


#-----------------------------------------------------------------------------
------

#Name       : ORACLE_HOME

#Datatype   : String

#Description: Complete path of the Oracle Home.

#Example: ORACLE_HOME = "C:\OHOME1"

#-----------------------------------------------------------------------------
------

ORACLE_HOME="/nfs/users/huchchen/OraHome_8"


#-----------------------------------------------------------------------------
------

#Name       : ORACLE_BASE

#Datatype   : String

#Description: Complete path of the Oracle Base.

#Example: ORACLE_BASE = "C:\app"

#-----------------------------------------------------------------------------
------

ORACLE_BASE=<Value Unspecified>


#-----------------------------------------------------------------------------
------
```

```
#Name        : ORACLE_HOME_NAME

#Datatype    : String

#Description: Oracle Home Name. Used in creating folders and services.

#Example: ORACLE_HOME_NAME = "OHOME1"

#-----------------------------------------------------------------------
------

ORACLE_HOME_NAME="OUIHome8"


#-----------------------------------------------------------------------
------

#Name        : SHOW_WELCOME_PAGE

#Datatype    : Boolean

#Description: Set to true if the Welcome page in OUI needs to be shown.

#Example: SHOW_WELCOME_PAGE = false

#-----------------------------------------------------------------------
------

SHOW_WELCOME_PAGE=true


#-----------------------------------------------------------------------
------

#Name        : SHOW_CUSTOM_TREE_PAGE

#Datatype    : Boolean

#Description: Set to true if the custom tree page in OUI needs to be shown.

#Use this page to select or de-select dependencies. This page appears only
in a custom install type.

#Example: SHOW_CUSTOM_TREE_PAGE = false

#-----------------------------------------------------------------------
------

SHOW_CUSTOM_TREE_PAGE=true
```

```
#------------------------------------------------------------------------------
------
#Name        : SHOW_COMPONENT_LOCATIONS_PAGE
#Datatype   : Boolean
#Description: Set to true if the component locations page in OUI needs to
be shown.
#This page only appears if there are products whose installed directory can
be changed.
#If you set this to false you will prevent the user from being able to
specify alternate directories.
#Example: SHOW_COMPONENT_LOCATIONS_PAGE = false
#------------------------------------------------------------------------------
------

SHOW_COMPONENT_LOCATIONS_PAGE=true


#------------------------------------------------------------------------------
------
#Name        : SHOW_SUMMARY_PAGE
#Datatype   : Boolean
#Description: Set to true if the summary page in OUI needs to be shown.
#The summary page shows the list of components that will be installed in
this session.
#Example: SHOW_SUMMARY_PAGE = true
#------------------------------------------------------------------------------
------

SHOW_SUMMARY_PAGE=true


#------------------------------------------------------------------------------
------
```

```
#Name       : SHOW_INSTALL_PROGRESS_PAGE

#Datatype   : Boolean

#Description: Set to true if the install progress page in OUI needs to be
shown.

#This page shows the current status in the installation. The current status
includes the product being installed and the file being copied.

#Example: SHOW_INSTALL_PROGRESS_PAGE = true

#------------------------------------------------------------------------
------

SHOW_INSTALL_PROGRESS_PAGE=true


#------------------------------------------------------------------------
------

#Name       : SHOW_REQUIRED_CONFIG_TOOL_PAGE

#Datatype   : Boolean

#Description: Set to true if the required config assistants page in OUI needs
to be shown.

#This page shows the list of required configuration assistants that are part
of this installation.

#It shows the status of each assistant, including any failures with detailed
information on why it failed.

#Example: SHOW_REQUIRED_CONFIG_TOOL_PAGE = true

#------------------------------------------------------------------------
------

SHOW_REQUIRED_CONFIG_TOOL_PAGE=true


#------------------------------------------------------------------------
------

#Name       : SHOW_CONFIG_TOOL_PAGE

#Datatype   : Boolean
```

#Description: Set to true if the config assistants page in OUI needs to be shown.

#This page shows the list of configuration assistants that are part of this installation and are configured to launch automatically.

#It shows the status of each assistant, including any failures with detailed information on why it failed.

#Example: SHOW_CONFIG_TOOL_PAGE = true

#--

SHOW_CONFIG_TOOL_PAGE=true

#--

#Name   : SHOW_RELEASE_NOTES

#Datatype   : Boolean

#Description: Set to true if the release notes of this installation need to be shown at the end of installation.

#This dialog is launchable from the End of Installation page and shows the list of release notes available for the products just installed.

This also requires the variable SHOW_END_SESSION_PAGE variable to be set to true.

#Example: SHOW_RELEASE_NOTES = true

#--

SHOW_RELEASE_NOTES=true

#--

#Name   : SHOW_ROOTSH_CONFIRMATION

#Datatype   : Boolean

```
#Description: Set to true if the Confirmation dialog asking to run the
root.sh script in OUI needs to be shown.

#Valid only for Unix platforms.

#Example: SHOW_ROOTSH_CONFIRMATION = true

#-------------------------------------------------------------------------
------

SHOW_ROOTSH_CONFIRMATION=true


#-------------------------------------------------------------------------
------

#Name         : SHOW_END_SESSION_PAGE

#Datatype   : Boolean

#Description: Set to true if the end of session page in OUI needs to be
shown.

#This page shows if the installation is successful or not.

#Example: SHOW_END_SESSION_PAGE = true

#-------------------------------------------------------------------------
------

SHOW_END_SESSION_PAGE=true


#-------------------------------------------------------------------------
------

#Name         : SHOW_EXIT_CONFIRMATION

#Datatype   : Boolean

#Description: Set to true if the confirmation when exiting OUI needs to be
shown.

#Example: SHOW_EXIT_CONFIRMATION = true

#-------------------------------------------------------------------------
------

SHOW_EXIT_CONFIRMATION=true
```

```
#------------------------------------------------------------------------
------
#Name       : NEXT_SESSION

#Datatype   : Boolean

#Description: Set to true to allow users to go back to the File Locations
page for another installation. This flag also needs to be set to true in
order to process another response file (see NEXT_SESSION_RESPONSE).

#Example: NEXT_SESSION = true

#------------------------------------------------------------------------
------

NEXT_SESSION=false


#------------------------------------------------------------------------
------

#Name       : NEXT_SESSION_ON_FAIL

#Datatype   : Boolean

#Description: Set to true to allow users to invoke another session even if
current install session has failed. This flag is only relevant if
NEXT_SESSION is set to true.

#Example: NEXT_SESSION_ON_FAIL = true

#------------------------------------------------------------------------
------

NEXT_SESSION_ON_FAIL=true


#------------------------------------------------------------------------
------

#Name       : NEXT_SESSION_RESPONSE

#Datatype   : String
```

```
#Description: Set to true to allow users to go back to the File Locations
page for another installation. This flag also needs to be set to true in
order to process another response file (see NEXT_SESSION_RESPONSE).

#Example: NEXT_SESSION_RESPONSE = "nextinstall.rsp"

#------------------------------------------------------------------------
------

NEXT_SESSION_RESPONSE=<Value Unspecified>


#------------------------------------------------------------------------
------

#Name       : DEINSTALL_LIST

#Datatype   : StringList

#Description: List of components to be deinstalled during a deinstall
session.

#Example: DEINSTALL_LIST = {"workbench","12.1.3.0.0"}

#------------------------------------------------------------------------
------

DEINSTALL_LIST={"workbench","12.1.3.0.0"}


#------------------------------------------------------------------------
------

#Name       : SHOW_DEINSTALL_CONFIRMATION

#Datatype   : Boolean

#Description: Set to true if deinstall confimation is needed during a
deinstall session.

#Example: SHOW_DEINSTALL_CONFIRMATION = true

#------------------------------------------------------------------------
------

SHOW_DEINSTALL_CONFIRMATION=true
```

```
#--------------------------------------------------------------------------
------
#Name       : SHOW_DEINSTALL_PROGRESS
#Datatype   : Boolean
#Description: Set to true if deinstall progress is needed during a deinstall
session.
#Example: SHOW_DEINSTALL_PROGRESS = true
#--------------------------------------------------------------------------
------
SHOW_DEINSTALL_PROGRESS=true


#--------------------------------------------------------------------------
------
#Name       : CLUSTER_NODES
#Datatype   : StringList
#Description: This variable represents the cluster node names selected by
the user for installation.
#Example: CLUSTER_NODES = {"node1","node2"}
#--------------------------------------------------------------------------
------
CLUSTER_NODES={}


#--------------------------------------------------------------------------
------
#Name       : ACCEPT_LICENSE_AGREEMENT
#Datatype   : Boolean
#Description: By setting this variable to true, you are accepting the
license agreement. This variable is used only for silent installations.
#Example: ACCEPT_LICENSE_AGREEMENT = true
```

```
#------------------------------------------------------------------------
------

ACCEPT_LICENSE_AGREEMENT=false


#------------------------------------------------------------------------
------

#Name       : METALINK_LOCATION

#Datatype   : String

#Description: This variable represents the Oracle metalink location.

#------------------------------------------------------------------------
------

METALINK_LOCATION=<Value Unspecified>


#------------------------------------------------------------------------
------

#Name       : METALINK_USERNAME

#Datatype   : String

#Description: This variable represents the Oracle metalink user name.

#------------------------------------------------------------------------
------

METALINK_USERNAME=<Value Unspecified>


#------------------------------------------------------------------------
------

#Name       : METALINK_PASSWORD

#Datatype   : String

#Description: This variable represents the corresponding Oracle metalink
password.

#------------------------------------------------------------------------
------
```

```
METALINK_PASSWORD=<Value Unspecified>


#-----------------------------------------------------------------------
------
#Name        : PROXY_HOST
#Datatype   : String
#Description: The proxy host  used to connect to Oracle metalink.
#Example: PROXY_HOST =
#-----------------------------------------------------------------------
------
PROXY_HOST=""


#-----------------------------------------------------------------------
------
#Name        : PROXY_PORT
#Datatype   : String
#Description: The proxy port used to connect to Oracle metalink.
#Example: PROXY_PORT =
#-----------------------------------------------------------------------
------
PROXY_PORT=""


#-----------------------------------------------------------------------
------
#Name        : PROXY_REALM
#Datatype   : String
#Description: The realm for the proxy used to connect to Oracle metalink.
#Example: PROXY_REALM =
```

```
#------------------------------------------------------------------------
------
PROXY_REALM=<Value Unspecified>


#------------------------------------------------------------------------
------
#Name       : PROXY_USER
#Datatype   : String
#Description: The username for the proxy  used to connect to Oracle metalink.
#Example: PROXY_USER =
#------------------------------------------------------------------------
------
PROXY_USER=""


#------------------------------------------------------------------------
------
#Name       : PROXY_PWD
#Datatype   : String
#Description: The password for the proxy  used to connect to Oracle metalink.
#Example: PROXY_PWD =
#------------------------------------------------------------------------
------
PROXY_PWD=<Value Unspecified>


#------------------------------------------------------------------------
------
#Name       : DONT_PROXY_FOR
#Datatype   : String
#Description: The dont proxy for list.
```

```
#Example: DONT_PROXY_FOR =

#-------------------------------------------------------------------------------
------

DONT_PROXY_FOR=<Value Unspecified>



#-------------------------------------------------------------------------------
------

#Name        : TOPLEVEL_COMPONENT

#Datatype    : StringList

#Description: The top level component to be installed in the current
session.

#Example: TOPLEVEL_COMPONENT = {"workbench","12.1.3.0.0"}

#-------------------------------------------------------------------------------
------

TOPLEVEL_COMPONENT={"workbench","12.1.3.0.0"}



#-------------------------------------------------------------------------------
------

#Name        : SHOW_SPLASH_SCREEN

#Datatype    : Boolean

#Description: Set to true if the initial splash screen in OUI needs to be
shown.

#Example: SHOW_SPLASH_SCREEN = true

#-------------------------------------------------------------------------------
------

SHOW_SPLASH_SCREEN=true



#-------------------------------------------------------------------------------
------

#Name        : SELECTED_LANGUAGES
```

```
#Datatype   : StringList

#Description: Languages in which the components will be installed.

#Component  : workbench

#-----------------------------------------------------------------------------
------


SELECTED_LANGUAGES={"en"}


#-----------------------------------------------------------------------------
------

#Name       : COMPONENT_LANGUAGES

#Datatype   : StringList

#Description: Languages supported by this component.List of supported
languages : {"en","ja"}

#Component  : workbench

#-----------------------------------------------------------------------------

COMPONENT_LANGUAGES={"en"}

#-----------------------------------------------------------------------------
------

#Name       : SAMPLES_INSTALL

#Datatype   : Boolean

#Description: true if install samples. otherwise false.

#Component  : workbench

#-----------------------------------------------------------------------------
------

SAMPLES_INSTALL=true


#-----------------------------------------------------------------------------
------
```

```
#Name       : OVER_WRITE

#Datatype   : Boolean

#Description: overwrite if found same version on this ORACLE_HOME

#Component  : workbench

#-------------------------------------------------------------------------
------


OVER_WRITE=true
```

Performing Post-Installation Tasks

The following sections describe the tasks you perform after installing Oracle Tuxedo Application Rehosting Workbench:

- Understanding the Oracle Tuxedo Application Rehosting Workbench Directory Structure

- Installation Verification

- Uninstalling Oracle Tuxedo Application Rehosting Workbench

Understanding the Oracle Tuxedo Application Rehosting Workbench Directory Structure

During the Oracle Tuxedo Application Rehosting Workbench software installation, all files are decompressed within the installation directory.

Table 4-1 lists Oracle Tuxedo Application Rehosting Workbench directories and files under installed directory `$ORACLE_HOME/artwb12.1.3.0.0`.

Table 4-1 Oracle Tuxedo Application Rehosting Workbench Directory and File Structure

Directory Name	Description
`refine`	Oracle Tuxedo Application Rehosting Workbench directory.
`refine/scripts`	Utilities which are not platform dependent.
`refine/scripts/lib`	Perl libraries.

Table 4-1 Oracle Tuxedo Application Rehosting Workbench Directory and File Structure

`refine/Linux64`	Platform dependent binaries and utilities for Linux 64-bit.
`refine/Linux32`	Platform dependent binaries and utilities for Linux 32-bit.
`refine/convert-data`	Data translator scripts and templates.
`refine/gcf`	Configuration files for Oracle Tuxedo Application Rehosting Workbench control.
`fixed-copy`	COBOL copy books which are necessary for compilation.
`samples`	Oracle Tuxedo Application Rehosting Workbench samples. Optional install.

Installation Verification

Two sample applications are provided to verify the installation and also to demonstrate the configuration. The Readme file in the samples provides guidance on how to execute the samples.

Before executing Oracle Tuxedo Application Rehosting Workbench, Oracle recommends that you set umask value as strict as possible.

Uninstalling Oracle Tuxedo Application Rehosting Workbench

Uninstalling Oracle Tuxedo Application Rehosting Workbench in GUI-Mode

The following steps show uninstallation process in GUI mode.

1. Invoke OUI introduction panel

   Go to `<ORACLE_HOME>/oui/bin`, run `"runInstaller"` on Unix/Linux platform.

2. Uninstalling panel

   Select products and click "Remove" to remove those products from your system.

3. Confirm panel

   To remove the products, click "Yes" in the following Confirmation dialog. Click "No" to cancel the removal.

4. Uninstall complete panel

   After uninstall finishes, this panel appears and reports whether it is successful.

5. Inventory panel

   Check the product list from Inventory, finding if the product is already uninstall successfully.

Uninstalling Oracle Tuxedo Application Rehosting Workbench in Silent Mode

Not only can you perform command line installations, you can also perform command line deinstallations. A command line deinstallation enables you to remove Oracle products or Oracle homes from your system without using the Oracle Universal Installer graphical user interface.

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

Use the following commands to immediately display the Inventory dialog box, which allows you to select items for removal without navigating the Oracle Universal Installer startup screen:

```
./runInstaller -deinstall -silent (on UNIX)
```

If you want to hide the inventory dialog box during a deinstallation, you can specify the products to be removed in the DEINSTALL_LIST parameter of the response file; specify Oracle homes to be removed with the REMOVE_HOMES variable; specify the name of the current Oracle home name with ORACLE_HOME_NAME. For example, on a UNIX machine, enter:

```
./runInstaller -deinstall -silent
"DEINSTALL_LIST={"workbench","12.1.3.0.0"}" ORACLE_HOME_NAME="OUIHome2"
REMOVE_HOMES={"/home/test/OraHome_2"}
```

Oracle Tuxedo Application Rehosting Workbench Supported Platforms

Table A-1 lists Oracle Tuxedo Application Rehosting Workbench 12*c* Release 2 (12.1.3) supported platforms.

Table A-1 Oracle Tuxedo Application Rehosting Workbench 12c Release 2 (12.1.3) Supported Platforms

Platform	GA Port/ Post-GA Port & Certification	OS EOL Date
Oracle Linux 5.6 (64-bit) on x86-64	GA	TBD
Oracle Linux 6.0 (64-bit) on x86-64	GA	TBD
Red Hat Enterprise Linux 5.6 (64-bit) on x86-64	GA	TBD
Red Hat Enterprise Linux 6.0 (64-bit) on x86-64	GA	TBD

Supported Platform Data Sheets

Oracle Linux 5.6 (64-bit) on x86-64

The software requirements for Oracle Linux 5.6 (64-bit) on x86-64 are as follows:

- Perl v5.8.5 or above

Oracle Linux 6.0 (64-bit) on x86-64

The software requirements for Oracle Linux 6.0 (64-bit) on x86-64 are as follows:

● Perl v5.8.5 or above

Red Hat Enterprise Linux 5.6 (64-bit) on x86-64

The software requirements for Red Hat Enterprise Linux 5.6 (64-bit) on x86-64 are as follows:

● Perl v5.8.5 or above

Red Hat Enterprise Linux 6.0 (64-bit) on x86-64

The software requirements for Red Hat Enterprise Linux 6.0 (64-bit) on x86-64 are as follows:

● Perl v5.8.5 or above