Oracle® Applications

Maintenance Utilities

Release 11*i* (11.5.10.2)

Part No. B19300-01

July 2005

ORACLE®

Oracle Applications Maintenance Utilities, Release 11*i* (11.5.10.2)

Part No. B19300-01

Primary Author: Carole Eubanks and Yun Shaw

Contributing Author: Henriette Fux, Richard Lotero, MyXuan Ho

Contents

3 Configuration

4 Maintenance

5 Patching

6 Reporting

Index

Send Us Your Comments

Oracle Applications Maintenance Utilities, Release 11*i* (11.5.10.2)

Part No. B19300-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. 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 about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us at apps_relgrp_us@oracle.com.

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

Preface

Oracle Applications Maintenance Utilities and *Oracle Applications Maintenance Procedures* make up the *Maintaining Oracle Applications Documentation Set*. This manual provides information about the utilities used for installing, upgrading, and maintaining Oracle Applications products.Its companion, *Oracle Applications Maintenance Procedures*, describes how to maintain the Oracle Applications file system and the database.

> **Note:** Much of the information in both these books was contained in the *AD Utilities Guide* in earlier releases.

Intended Audience

This book is intended for database administrators and system administrators who are responsible for performing Oracle Applications maintenance tasks.

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 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 does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Structure

This book contains the following chapters:

Chapter 1

Provides general information about the AD utilities and the OAM utilities.

Chapter 2

Describes the way Rapid Install and AutoUpgrade work. Specific installation steps are in *Installing Oracle Applications*. Specific steps for completing an upgrade from Release 10.7 or 11.0 to this version of Release 11*i* are in *Upgrading Oracle Applications*.

Chapter 3

Describes the utilities used to configure a system, including changing the configuration, registering products, adding new products, and converting the character set.

Chapter 4

Describes the utilities you use to maintain your system. The primary maintenance utility is AD Administration.

Chapter 5

Describes the patching process. It includes a description of patch structure, and discusses merged patches, the AutoPatch utility, and Patch Wizard.

Chapter 6

Describes the reporting capabilities of various utilities, both AD and OAM.

Related Documents

This book was current as of the time it was initially published. It is included in the *Oracle Applications Document Library*, which is supplied in the Release 11*i* software bundle. Later versions of this and other documents that have changed sufficiently between releases to warrant re-publishing are made available at the following URL:

`http://www.oracle.com/technology/documentation/applications.html`

You can also purchase hard-copy documentation from the Oracle Store at `http://oraclestore.oracle.com`.

In addition to this book, we suggest you have the following additional references on hand.

If you are looking for...	See these documents...
Additional information	*Oracle Applications Concepts* *Upgrading Oracle Applications* *Installing Oracle Applications* *Maintaining Oracle Applications Documentation Set: Oracle Applications Maintenance Procedures* *Oracle Applications Installation Update Notes** *Oracle Applications Release Notes** *Oracle Applications NLS Release Notes** *Oracle Applications System Administrator's Guide - Configuration* *Oracle Applications System Administrator's Guide - Maintenance* *Oracle Applications System Administrator's Guide - Security* *Oracle Self-Service Web Applications Implementation Manual* *Oracle Workflow Administrator's Guide* *Oracle Workflow Developer's Guide* *Oracle Application Object Library/Workflow Technical Reference Manual*
Information on new features in this release	*About AD.I** *Release Content Documents and Features Summary Matrices** *Oracle Applications DBA 11i+ Features Matrix** *Oracle Applications Product Update Notes* *Electronic Technical Reference Manual (eTRM)**
Application-specific features	Oracle Applications user's guides Oracle Applications implementation manuals *Multiple Organizations in Oracle Applications* *Multiple Reporting Currencies in Oracle Applications*
Information about custom development	*Oracle Applications User Interface Standards for Forms-based Products* *Oracle Applications Developers' Guide*
Database information	Oracle9*i* Documentation Set

*Available only on Oracle*MetaLink*

Patch readme files may also contain information about new documentation that you can download.

Training and Support

Oracle offers a complete set of training courses and multi-level support services.

Training

You can attend training courses at any Oracle Education Center, arrange for trainers to teach at your facility, or use Oracle Learning Network (OLN) — Oracle University's online education utility. Oracle training professionals can also develop custom courses using your organization structure, terminology, and data as examples.

Support

The Oracle support team includes your Technical Representative and Account Manager. It also includes Oracle consultants and support specialists who have expertise in your business area, and in managing an Oracle RDBMS server and your hardware and software environment.

Oracle*MetaLink* is a self-service, web-based support connection, which is maintained by Oracle Support Services 24 hours a day, 7 days a week. Use it to obtain information and advice from technical libraries and forums, download patches, look at bug details, and create or update TARs. Register at http://metalink.oracle.com, and check for updates and information before you install or upgrade your Oracle Applications. The *Start Here* CD also contains links to the various resources on Oracle*MetaLink*.

Conventions

The following conventions are used in this book:

Convention	Meaning
UNIX: **Windows**:	Indicates platform-specific information. This guide contains information for both UNIX and Windows platforms. All instructions for UNIX platforms also apply to Linux platforms, unless otherwise noted.
$ or C:\>	Represents the platform-specific command prompt. Your prompt may differ.
`Monospace text`	Represents command line text. Type this text exactly as shown.
< >	Text enclosed in angle brackets represents a variable. Substitute a value for the variable text. Do not type the brackets
[ ]	Encloses optional items or indicate a function key. Do not type the brackets.
|	Represents an OR option among several options. You must enter only one of the options. Do not type the vertical bar.
\	In examples of commands you type online, a backslash at the end of a line signifies that you must type the entire command on one line. *Do not type the backslash*.
Special notes	Additional Information, Note, and Caution boxes alert you to particular information within the body of the book.

1

Getting Started

This chapter contains basic information about Oracle Applications maintenance utilities, both command line and web-based. It contains the following topics:

- System Maintenance Utilities
- AD Command Line Utilities
- Using Parallel Processing
- OAM Web-based Utilities

System Maintenance Utilities

You use Oracle Applications system maintenance utilities to perform a variety of operations from installing and upgrading Oracle Applications systems, to updating configuration parameters, to maintaining and patching your database and file system, to producing system reports. In this book, we have categorized these utilities by the way you access and use them: from the command line and from a web-based interface.

> **Note:** Some of the functionality formerly provided by the utilities described in this book is now managed by AutoConfig. If you have not enabled AutoConfig in your system, you must do so in order to use the information in this book.

Command Line Utilities

The utilities generally referred to as AD utilities (AD is an abbreviation for Applications DBA) are started and run from the command line. They kick off processes that perform tasks such as applying and merging patches, upgrading products, and initiating a variety of system maintenance tasks. As they run, the utilities prompt you for system-specific parameters necessary to perform the maintenance task. In addition, many of the utilities produce reports, which contain information such as job timing and file versions.

The AD utilities have similar interfaces, operation, input, and report formats. Many also share the ability to accept arguments, flags, and options, which you can use to refine the actions they perform. You add the argument on the command line when you start the utility. For example, to specify the number of workers that AutoPatch should run in parallel when applying a patch, you indicate the number of worker processes on the command line when you start AutoPatch. A list of commonly used command line arguments and flags, and a brief description of how to use them, begins on page 1-9 in this chapter.

The AD utilities described in this book are shown in the following table.

AD Utility Name	Executable	Description
AD Administration	adadmin	Performs maintenance tasks for Oracle Applications.
AD Check Digest	adchkdig	Checks the integrity of Oracle Applications patches downloaded from Oracle*MetaLink*.
AD Configuration	adutconf.sql	Reports standard information about the installed configuration of Oracle Applications.
AD Controller	adctrl	Manages parallel workers in AD Administration, AutoPatch, and AutoUpgrade.
AD File Identification	adident	Reports the version and translation level of an Oracle Applications file.
AD File Character Set Converter	adncnv	Converts a file from one character set to another.
AD Merge Patch	admrgpch	Merges multiple patches into a single merged patch.
AD Relink	adrelink.sh	Relinks Oracle Applications executable programs with the Oracle server product libraries.
AD Splicer	adsplice	Adds off-cycle products.
AD Job Timing Report	adtimrpt.sql	Reports a summary of the timing for jobs run by parallel workers.
AutoPatch	adpatch	Applies patches and other system updates.
AutoUpgrade	adaimgr	Works in conjunction with Rapid Install to upgrade a system to the latest version of Oracle Applications.
Rapid Install	rapidwiz	Provides a wizard for entering parameters that are specific to a new installation or an upgrade of an Oracle Applications system.

Web-based Utilities

Oracle Applications Manager (OAM) provides a web-based interface where system administrators can monitor system status, administer services, examine system configuration, manage Oracle Workflow, view applied patches, and measure system usage. It provides a concise overview of the state of your Oracle Applications system, and serves as a gateway to utilities that allow you to manage your configuration, review patch history, determine patches that will bring your system up to date, register products and languages, and other maintenance activities.

> **Additional Information:** See Oracle Applications Manager in *Oracle Applications Concepts*. See also the Oracle Applications Manager Help function.

Some maintenance utilities discussed in this book, and in *Oracle Applications Maintenance Procedures*, are part of OAM. They are listed in the following table. Note that because of the web-based interface, there is no need for an executable (start command).

OAM Utility Name	Description
AutoConfig	Use to view current context files, edit parameters contained in the context files, view previous context files, and compare current context files against previous ones. Must be used in conjunction with the AutoConfig (command line) script to propagate changes to system configuration files.
License Manager	Registers additional Oracle Applications products, country-specific functionalities, or languages. You can also use License Manager to change the base language for your system.
Patch Wizard	Determines patches that have not been applied, but that should be applied to keep the system current. Downloads and merges patches from Oracle*Metalink*.
Applied Patches	Uses key patch information in the patch history database. You can search the database to create reports in several formats.

Online Help

Both the AD utilities and the OAM utilities provide a help function.

Command Line Help

For the AD command line utilities, you can request a list of arguments by typing the utility executable (start command) and adding *help=y*. For example, to access help for AutoPatch, type:

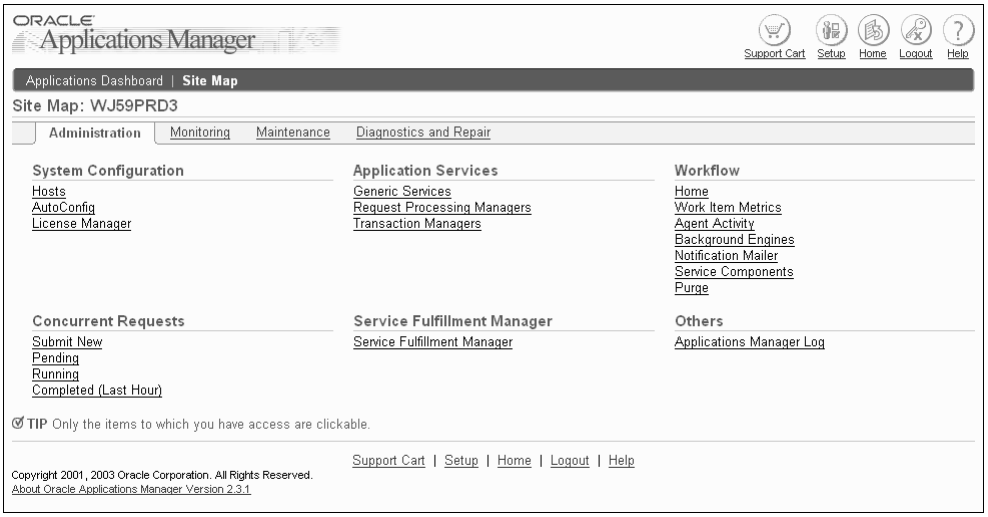
```
adpatch help=y
```

The arguments and options that you can use to refine the operation of a utility are listed, along with a brief description of how they work. Here's an excerpt from the command line help for AutoPatch.

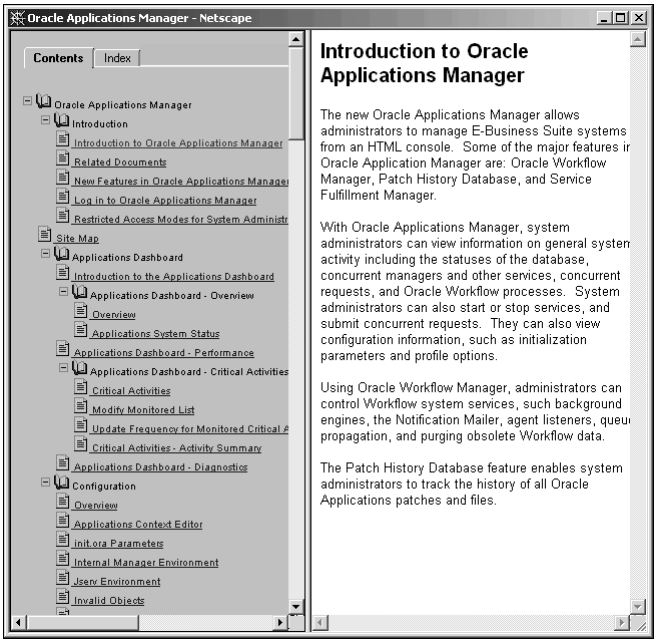
```
usage: adpatch [help=y]
       adpatch Pre-Install Mode
               [preinstall=y|n]
       adpatch Test Mode
               [apply=y|n]
       adpatch Non-Interactive mode
               [defaultsfile=<$APPL_TOP/admin/SID/defaultsfile>]
               [logfile=<logfile>] [interactive=y|n] [workers=<workers>]
               [patchtop=<patchtop>] [driver=<driver_file>] [restart=y|n]
               [abandon=y|n] [wait_on_failed_job=y|n]
        adpatch
               [localworkers=<localworkers>] [workers=<workers>]
               [printdebug=y|n] [parallel_index_threshold=<threshold_value>]
               [order=<order>]  [flags=<flags>] [options=<options>]
where
* help=y      - Prints help information about adpatch options.
* preinstall  - To run adpatch in Pre-Install Mode.
                Default - No.
* apply       - To run adpatch in Test Mode.
                Default - Yes.
* interactive - Invokes adpatch in Non-Interactive mode when
                "interactive=no" is specified.
                Default - Yes.
```

OAM Help

OAM Help is in the form of a web-based directory tree that lists overview information and specific features available from the Site Map or the Applications Dashboard. You access the menu by clicking the Help icon from any page in the Oracle Applications Manager. For example, from the Site Map Administration tab, click the Help icon.



The OAM help directory appears in a separate window. Click either the Contents tab or Index tab to expand the branches and view the entire tree.



Click a plus sign to see more topics, and click any individual link to access information on that topic.

Individual topics may include topical essays, procedures, and page descriptions. The help associated with the utilities and features discussed in this book provides navigation paths, field definitions, and general information about using the page.

AD Command Line Utilities

The AD maintenance utilities were developed to perform specific Applications maintenance and reporting tasks from the command line. For example, you use AutoPatch to apply all types of patches to your system, you use AutoUpgrade to upgrade your Applications products from one release level, and you use AD Administration to perform routine maintenance tasks.

However, even though each utility has a specialized function, they are designed to complement each other, therefore many (and often all) employ some similar operations. This section summarizes the operations that AD utilities have in common. Subsequent chapters describe each utility's unique characteristics in detail.

> **Additional Information:** See *Oracle Applications Maintenance Procedures* for specific tasks performed using the AD utilities.

Common AD Operations

AD Administration, AutoUpgrade, and AutoPatch, as well as many of the other AD utilities, employ similar features and operations as they perform processing tasks. For example, most rely on prompts to gather values for system-specific processes, and all automatically create log files to record processing actions. This section describes these common operations.

Prompts

Many AD utilities prompt for information necessary for completing a task. Prompts typically include a description of the information needed, and may include a default answer (in square brackets). For example:

```
The ORACLE username specified below for Application Object Library uniquely
identifies your existing product group: APPLSYS

Enter the ORACLE password of Application Object Library [APPS] :
```

Press [Return] to accept the default value, or type a new value after the colon and press [Return]. Read the prompts carefully to make sure you supply the correct information.

Interactive and Non-interactive Processing

The AD utilities perform processing tasks *interactively* by default. That means the utility prompts for system-specific information at the point where it needs it, making it necessary for you to be present during the entire operation in order to respond to the prompts.

AD Administration, AutoPatch, and AD Controller can run some file system and database tasks *non-interactively* — you store necessary information in a *defaults file*. The utility reads the information from this file rather than prompting you for the input you would typically supply when responding to prompts. Non-interactive processing is useful for scheduling routine tasks that require little or no user intervention.

> **Additional Information:** See Performing Maintenance Tasks Non-interactively and Performing Patching Tasks Non-interactively in *Oracle Applications Maintenance Procedures*. See also Monitoring and Controlling Parallel Processes in this chapter for information about running AD Controller non-interactively.

Log Files

All AD utilities record their processing actions and any errors that they encounter in log files. Many utilities prompt you for the name of the log file that will record the processing session.

```
<utility name> records your <utility name> session in a text file you specify.
Enter your <utility name> log file name or press [Return] to accept the default
name shown in brackets.

Filename [<utility name>.log] :
```

The default file name is <utility name>.log. For example, for AD Administration, the default log file is adadmin.log, and for AutoPatch, it is adpatch.log.

AD Administration, AutoPatch, and AutoUpgrade place the log file in $APPL_TOP/admin/<SID>/log, where <SID> is the value of the ORACLE_SID or TWO_TASK variable (UNIX), or in %APPL_TOP%\admin\<SID>\log, where <SID> is the value of the LOCAL variable (Windows). Log files created by Rapid Install and AutoConfig are located in the $APPL_TOP/admin/<CONTEXT_NAME>/log directory.

Other utilities may not prompt you for a log file name. Typically, they write the log file in the directory from which the utility was run. The discussion of each utility in this book includes information on where the log file is written.

Restart Files

Restart files contain information about what processing has already been completed. They are located in $APPL_TOP/admin/<SID>/restart (UNIX) or in %APPL_TOP%\admin\<SID>\restart (Windows).

If a utility stops during processing due to an error, or you use AD Controller (in the case of parallel processing) to shut down workers while they are performing processing tasks, you can restart the utility. If you do, it looks for restart files to determine if there was a previous session. If the files exist, the utility prompts you to continue where the processing left off, or to start a new process. If you choose to continue, it reads the restart files to see where the process left off, and continues the process from that point.

> **Caution:** Do not modify or delete any manager or worker restart files unless specifically told to do so by Oracle Support Services.

By default, AD utilities delete their restart files when processing is complete, but leave backup versions with the extensions .bak, .bk2, or .bk3.

> **Caution:** Restart files record passwords to your Oracle Applications products. You should restrict access to all restart files (located in $APPL_TOP/admin/<SID>/restart). If you are running a utility with *options=nohidepw*, the log files may also contain passwords on lines prefixed with HIDEPW.

Configuration and Environment Files

Most AD utilities require access to system parameters stored in various configuration and environment files when processing maintenance tasks. For example, it may be

necessary to know the location of the iAS ORACLE_HOME or the RDBMS ORACLE_HOME.

Configuration and environment files are generated by AutoConfig during an installation or upgrade. You typically do not have to manually update or maintain the information in these files. They are updated when you run the AutoConfig script.

> **Additional Information:** See AutoConfig in *Oracle Applications Concepts*. See also AutoConfig in Chapter 3.

The following table lists configuration and environment files commonly used by the AD command line utilities, and in some cases, by the OAM web-based utilities. Note that <CONTEXT_NAME> defaults to <SID>_<hostname>.

File name	Location	Description
adconfig.txt	APPL_TOP/admin	Contains environment information used by all AD utilities. *Do not* update this file manually.
<CONTEXT_NAME>.env (UNIX) <CONTEXT_NAME>.cmd (Windows)	iAS ORACLE_HOME	Used to configure the environment when performing maintenance operations on the iAS ORACLE_HOME.
<CONTEXT_NAME>.env (UNIX) <CONTEXT_NAME>.cmd (Windows)	RDBMS ORACLE_HOME	Used to configure the environment when performing maintenance operations on the database.
APPS<CONTEXT_NAME>.env (UNIX) APPS<CONTEXT_NAME>.cmd (Windows)	APPL_TOP	Named APPSORA in earlier releases, this file calls the environment files needed to set up the APPL_TOP and the 8.0.6 ORACLE_HOME.
<CONTEXT_NAME>.env (UNIX) <CONTEXT_NAME>.cmd (Windows)	APPL_TOP	Called by APPS<CONTEXT_NAME>.env (UNIX) or APPS<CONTEXT_NAME>.cmd (Windows) file to set up the APPL_TOP. (This file calls adovars.env or adovars.cmd.)
<CONTEXT_NAME>.env (UNIX) <CONTEXT_NAME>.cmd (Windows)	8.0.6 ORACLE_HOME	Called by APPS<CONTEXT_NAME>.env (UNIX) or APPS<CONTEXT_NAME>.cmd (Windows) file to set up the 8.0.6 ORACLE_HOME.
adovars.env (UNIX) adovars.cmd (Windows)	APPL_TOP/admin	Called by the <CONTEXT_NAME>.env (UNIX) or <CONTEXT_NAME>.cmd (Windows) file located in the APPL_TOP. Used to set environment variables for Java and HTML.

The following configuration and environment files are also used by most AD utilities, but are not created by AutoConfig. Do not update these files manually.

File name	Location	Description
applora.txt	APPL_TOP/admin	Contains information about required init.ora parameters for runtime.
applorau.txt	APPL_TOP/admin	Contains information about required init.ora parameters for install and upgrade.
applprod.txt	APPL_TOP/admin	The AD utilities product description file, used to identify all products and product dependencies.
applterr.txt	APPL_TOP/admin	The AD utilities territory description file. It contains information on all supported territories and localizations.

File name	Location	Description
fndenv.env	FND_TOP	Sets additional environment variables used by Oracle Application Object Library. The default values should be applicable for all customers.

Feature Versions

In order to use some AD Administration and AutoPatch features, the version number of the feature must be the same in both the file system and the database. There may be times when these feature versions do not match. For example, if an AD patch did not run successfully to completion, it may have updated the file system, but not the database. In this case, the file system version and the database version could be different.

When you start AD Administration or AutoPatch, an information matrix scrolls on the screen. It indicates the status (Active=<Yes or No>) and version numbers of the following features: CHECKFILE, PREREQ, CONCURRENT_SESSIONS, PATCH_HIST_IN_DB, PATCH_TIMING, and SCHEMA_SWAP.

> **Additional Information:** OracleMetaLink document 174436.1 contains more information about feature versions implementation.

The matrix is for information only. No action is required unless the feature versions do not match. If they do not, you can use the OAM Applied Patches utility to determine which patches were applied successfully and verify the version level.

> **Additional Information:** See Applied Patches Information in Chapter 5.

The AD Interface

Some AD utilities are designed to perform a single function. For example, you run AutoPatch to apply patches to your system, or AD Merge Patch to merge separate patches into a single patch. These utilities do not use menus or input screens. All user interaction is from the command line in the form of prompts.

However, other utilities have multiple functions, which are presented on menus or input screens. For example, when you run AD Administration, the first screen you see is the main menu.

```
          AD Administration Main Menu
  -------------------------------------------------

  1.    Generate Applications Files menu

  2.    Maintain Applications Files menu

  3.    Compile/Reload Applications Database Entities menu

  4.    Maintain Applications Database Entities menu

  5.    Change Maintenance Mode

  6.    Exit AD Administration









Enter your choice [6] : 
```

From this screen, choose one of the submenus, and then from that menu, choose the process you want to run. Besides AD Administration, AD Controller and AutoUpgrade also display menus.

Command Line Arguments

You can direct the way the AD utilities operate by adding modifiers to the utility's start command. These modifiers may be in the form of arguments, flags, or options. They all refine the actions performed by a utility.

Command line arguments, flags, and options are in the "token=value" format, where *token* is the name of the modifier. You should enter both the argument and the value in lowercase type (the utility automatically converts the "token" portion to lowercase, but it cannot convert the "value").

For example:

```
$ adpatch LOGFILE=TEST.LOG
```

The token ("LOGFILE") will be converted to lowercase, but the value (TEST.LOG) is not recognized by the utility. The correct way to enter this command is:

```
$ adpatch logfile=test.log
```

You can enter more than one token=value argument on a single command line by separating them with one blank space as in the following AutoPatch command.

```
$ adpatch printdebug=y flags=hidepw
```

In some cases, you can include more than one value for a token. In this case, separate the values with commas. For example:

```
$ adpatch flags=nohidepw,trace
```

Comma-separated lists must not contain blank spaces. For example, this command is not valid:

```
$ adpatch flags=nohidepw, trace
```

Some command line arguments are used by several utilities and are listed in the following table. Other arguments are used only for a specific utility. For example, AutoPatch makes extensive use of command line arguments and options that are unique to that utility. They are listed and discussed separately in Chapter 5.

abandon	**Description**
Used by	AD Administration, AutoPatch
Purpose	Tells AD utilities to abandon an existing non-interactive session. Can be used only when interactive=n is also specified.
Values	y or n
Default	n, meaning that the last utility run non-interactively did not successfully complete the processing.
Example	adpatch interactive=n abandon=y

defaultsfile	**Description**
Used by	AD Administration, AutoPatch, AD Controller

defaultsfile	Description
Purpose	Specifies the defaults file which stores answers to interactive AD utility questions. Normally used non-interactively.
Values	A fully-qualified filename. Must be under the $APPL_TOP/admin/<SID> directory.
Default	None, meaning that no defaults file is used.
Example	adpatch defaultsfile=/d1/apps/prodappl/admin/prod1/prod_def.txt

help	Description
Used by	All AD utilities
Purpose	Summarizes available command line options.
Values	y or n
Default	n
Example	adpatch help=y

interactive	Description
Used by	AD Administration, AutoPatch, AD Controller
Purpose	Tells AD utilities whether to run either interactively or non-interactively.
Values	y or n
Default	y, meaning that the utility runs interactively.
Example	adpatch interactive=n

localworkers	Description
Used by	AD Administration, AutoPatch
Purpose	Specifies the number of workers to run on the primary node in a Distributed AD environment.
Values	1 to the maximum supported by your database, but not more than 999, inclusive
Default	Defaults to the value of the workers argument, which means all workers run on the primary node.
Example	adpatch workers=8 localworkers=3

logfile	Description
Used by	All AD Utilities
Purpose	Tells AD utilities what log file to use. Normally used when running a utility non-interactively.
Values	A file name (not a fully-qualified path name)
Default	None, meaning that the utility will prompt for the log file name.
Example	adpatch logfile=test.log

menu_option	Description
Used by	AD Administration, AD Controller
Purpose	When running one of these utilities non-interactively, used to connect the actions in a defaults file with a specific menu item.
Values	See list of menu options in the description of these utilities. Must be used with interactive=n and defaultsfile=<name of defaults file>.
Default	N/A
Example	adctrl interactive=n defaultsfile=$APPL_TOP/admin/prod/ctrldefs.txt menu_option=SHOW_STATUS

parallel_index_threshold	Description
Used by	AD Administration, AutoPatch
Purpose	Specifies the number blocks in a table. If a table contains less blocks than the threshold setting, indexes are created with parallel workers and serial DML. If the table contains more blocks than the threshold setting, indexes are created with one worker and parallel DML.
Values	0 to 2147483647; if set to 0, indexes are created with parallel workers and serial DML
Default	20000; meaning a threshold of 20,000 blocks.
Example	adpatch parallel_index_threshold=15000

printdebug	Description
Used by	All AD Utilities
Purpose	Tells AD programs to display extra debugging information. In some cases, the amount of extra debugging information is substantial.
Values	y or n
Default	n
Example	adpatch printdebug=y

restart	Description
Used by	AD Administration, AutoPatch, AD Controller
Purpose	Tells AD utilities running non-interactively to restart an existing session. Only valid when interactive=n is also specified.
Values	y or n
Default	n, meaning that the utility running non-interactively will expect to run a completely new session.
Example	adpatch interactive=n restart=y

wait_on_failed_job	Description
Used by	AD Administration, AutoPatch
Purpose	Directs the utilities to wait for user input in a non-interactive session when a job fails.
Values	y or n
Default	n
Example	adpatch wait_on_failed_job=yes

workers	Description
Used by	AD Administration, AutoPatch
Purpose	Specifies the number of workers to run. Normally used when running the utility non-interactively.
Values	1 to the maximum supported by your database, but not more than 999
Default	No, meaning that the program prompts for the number of workers to run.
Example	adpatch workers=8

AD Flags

The *flags*= argument is used by all AD utilities. It passes one of several generic flags to the utility. Enter one flag or a comma-separated list of flags. The default is None.

hidepw	Description
Default	hidepw
Purpose	Directs the utilities to either hide or show passwords in AD Utility log files.
Comments	By default, lines in an AD utility log file containing passwords are modified to hide the passwords. When nohidepw is specified, each line containing hidden passwords is followed by a corresponding line prefixed with HIDEPW:, showing the original line with passwords.
Example	adpatch flags=nohidepw

logging	Description
Default	logging (AutoPatch); nologging (AutoUpgrade)
Purpose	Tells the AD utility whether to create indexes using logging or nologging.

logging	Description
Comments	Using flags=nologging when creating indexes may increase performance. However, flags=nologging makes database media recovery incomplete and does not work with standby databases. Logging is the default in AutoPatch to support database media recovery and standby databases. We do not recommend using flags=nologging for production systems unless you make a complete backup both before and after running AutoPatch. Nologging is the default in AutoUpgrade because you must make a full database backup before and after the upgrade. flags=nologging affects indexes created through ODF only, not SQL scripts. The XDF utility always creates indexes with logging.
Example	adaimgr flags=logging

trace	Description
Default	notrace
Purpose	Tells the AD utility whether to log all database operations to a trace file.
Comments	RDBMS trace files created while running an AD utility may aid debugging. The flags=trace option creates multiple trace files for the AD utility and the AD workers. A new trace file is created each time the AD utility or a worker reconnects to the database. Note that flags=trace only traces database operations internal to the AD utility itself. Database operations in SQL scripts or external programs run by the AD utility are not recorded by flags=trace.
Example	adpatch flags=trace

AD utilities accept other arguments. However, they should be used only when instructed to do so by Oracle Support Services.

Running AD Utilities

To run AD utilities, set the environment to define the system configuration parameters. For example, a utility may require the directory path to the 8.0.6 ORACLE_HOME. This parameter, and others, make up your system environment.

> **Note:** Before setting the environment, Windows users must also configure Windows services.

> **Additional Information:** See Configuration and Environment Files on page 1-6.

Once you have pointed the utility to the correct environment, you start it by entering the executable (start command).

Setting the Environment

To set the Applications environment, complete the following steps. See the *Oracle Installation Update Notes* for any additional platform-specific steps.

1. Log in as applmgr (Applications file system owner).

2. Run the environment (UNIX) or command (Windows) file for the current APPL_TOP and database.

   UNIX:

   The environment file is typically APPS<CONTEXT_NAME>.env, and is located under APPL_TOP. From a Bourne, Korn, or Bash shell, type the following:

   ```
   $ . APPS<CONTEXT_NAME>.env
   ```

   Windows:

   Run %APPL_TOP%\envshell.cmd using either Windows Explorer or the Run command from the Start menu. This creates a Command Prompt window that contains the required environment settings for Oracle Applications. Run all subsequent commands in this Command Prompt window.

3. If you have made any changes to the environment, check that it is correctly set by typing the following commands:

   UNIX:

   ```
   $ echo $TWO_TASK
   $ echo $ORACLE_HOME
   $ echo $PATH
   ```

   Windows:

   ```
   C:\> echo %LOCAL%
   C:\> echo %ORACLE_HOME%
   C:\> echo %PATH%
   C:\> echo %APPL_CONFIG%
   ```

   For UNIX, the ORACLE_HOME must be set to the proper database directory, and TWO_TASK or LOCAL must identify the correct database. For Windows, APPL_CONFIG must be set to <CONTEXT_NAME>.

4. Ensure that there is sufficient temporary disk space.

   You should have at least 50 MB in the temporary directories denoted by $APPLTMP, $APPLPTMP, and $REPORTS60_TMP (UNIX) or %APPLTMP%, %APPLPTMP%, and %REPORTS60_TMP% (Windows). You should also have space in the operating system's default temporary directory, which is usually /tmp or /usr/tmp (UNIX) or C:\temp (Windows).

5. If you are running an AD utility to relink or update Oracle Applications product files or modify Oracle Applications database objects, shut down the concurrent manager, Web server listeners, forms server listeners if the files are on a node that contains the associated servers. For example, if the files are on the node that contains the concurrent processing server, shut down the concurrent managers.

   Additional Information: See Administer Concurrent Managers in *Oracle Applications System Administrator's Guide - Configuration*

6. Enable Maintenance mode if the maintenance task requires system downtime.

   Additional Information: See Changing Maintenance Mode in Chapter 4.

Configuring Windows Services

If you are running AD utilities on a Windows platform, you must first shut down all forms services, Web listener services, and concurrent manager services. In addition, you must verify that the database and database listeners are running.

To view and change the status of a service, follow these steps:

1. Select Start > Settings > Control Panel, and double-click on Services.
2. Highlight the appropriate service name and click Stop or Start as appropriate. The following table lists the services and status required when running an AD utility:

Service Type	Service Name	Status
Forms Services	OracleFormsServer-Forms60<SID>	Stopped
Web Services	Oracle Apache Server <SID>	Stopped
Concurrent Manager Services	OracleConcMgr<CONTEXTNAME>	Stopped
Apps Listener	Oracle<SID>Ora806TNSListener80APPS_<SID>	Started
Database Services	OracleService<SID>	Started
Database Listener	Oracle<SID>_<DB_VERS>_RDBMSTNSListener<SID>	Started

Starting a Utility

To start an AD utility, type the utility's executable name (start command) on the command line. For example, to start AD Administration, you would type:

```
$ adadmin
```

> **Additional Information:** See Command Line Utilities on page 1-1 for a list of AD executables.

Exiting or Stopping a Utility

When menu-driven utilities complete a processing task, they return you to the main menu, where you either choose another process or Exit. AD Administration is an example. Other utilities do not use a menu format. In this case, the utility exits automatically when processing is complete. AutoPatch, AD Merge Patch, and File Character Set Converter are examples.

Before it begins processing tasks, you can stop a utility by typing *abort* at any prompt. You can use this command only for utilities that display prompts, and only when a prompt is displayed on the screen.

In some cases, a utility may begin the processing actions, but quits before the actions are complete (because of an error). Or, during a parallel processing session, you may decide to stop the processing actions by shutting down the workers.

> **Additional Information:** See Troubleshooting in *Oracle Applications Maintenance Procedures* for more information about shutting down and restarting workers.

Restarting a Utility

You can restart a utility by entering the executable on the command line. When you restart, the utility prompts you to enter a new log file, or to specify the log file from the interrupted session. When you reuse the log file from a previous session, the utility adds the message "Start of <utility name> session" to the end of the file and appends messages from the continued session as it generates them.

The utility prompts you to do one of the following:

- Continue Session (the default)

  The utility checks the progress of the previous session in the restart files, and begins processing at the point where your last session stopped.

  Note: We recommend that you choose Continue Session. Some actions from the first session may be voided or duplicated if you start a new session.

- Start New Session

  The utility asks you to confirm your choice if you choose not to continue the previous session. It starts the process from the beginning.

If the process that stopped was running in parallel, a FND_INSTALL_PROCESSES table may exist. If it does, the utility asks if you want to drop the table. You must determine if AD Administration, AutoUpgrade, or AutoPatch is running in another session or on another node, wait until that session is complete before you start a new utility session on the current system.

Additional Information: See Managers on page 1-6.

Using Parallel Processing

In order to make better use of system resources, and to reduce the time it takes to complete certain maintenance tasks, AutoUpgrade, AutoPatch, and AD Administration have been designed to process jobs concurrently. This *parallel processing* makes use of *managers*, which direct the actions of *worker* processes. The manager assigns each worker a processing job, and monitors its progress. When a worker completes a job, the manager assigns it another, until all jobs are complete.

In addition, AD Administration and AutoPatch can be directed to distribute processing tasks across nodes in a multi-node system. This type of parallel processing operation is called *Distributed AD*. It further reduces time to complete a maintenance task by utilizing the processing capabilities of all the nodes in the system.

Note: You must have a shared application tier file system to use Distributed AD. See Sharing the Application Tier File System in Oracle Applications 11*i* (Oracle*MetaLink* document 233428.1).

Additional Information: See Distributing Processing Tasks in *Oracle Applications Maintenance Procedures*.

Processing Tasks in Parallel

Parallel processing is typically used by AD Administration, AutoPatch, and AutoUpgrade to:

- run database driver tasks, such as SQL scripts.
- generate various kinds of files, such as forms, reports, messages, and graphics files.

Workers complete processing tasks assigned to them by the manager. The utilities themselves determine the list of tasks to be performed and prioritize them for

execution. They also prompt for the number of workers to perform the tasks. For example, when AutoPatch is applying a database driver, it creates a list of database tasks and prompts you to specify the number of workers that should run concurrently to execute these tasks.

> **Note:** The worker processes are instances of the adworker program. This program can only be called by the manager processes, and cannot be run stand-alone.

Managers

The manager assigns each worker a unique ID and inserts a row for each worker in the FND_INSTALL_PROCESSES table. It creates this table to serve as a staging area for job information, and as a way to communicate with the worker. Communication is accomplished using two columns: CONTROL_CODE and STATUS.

The manager updates the table with a subset of the list of jobs, one job per worker. For example, if there are five workers, then the table holds five jobs (even though there may be 100 or more jobs involved in the complete action). The manager starts the workers and uses the CONTROL_CODE and STATUS columns to assign tasks. It polls these two columns continuously, looking for updates from the workers. As a worker finishes its assignment, the manager updates each row with the next task in the list, and leaves another message for the worker.

Once all jobs are complete, the manager tells the workers to shut down, and then drops the FND_INSTALL_PROCESSES table (after it is sure all workers have actually shut down).

Workers

Each worker updates the STATUS column, giving the manager a report on its progress. As the jobs are completed, the manager updates the table with the next job in the queue, and updates the CONTROL_CODE and STATUS columns telling the worker to start processing. If there is a failure, the worker reports a failed status.

For certain tasks, some worker processes spawn other *child processes* that do the actual work. The spawned child process returns a status code to the worker that spawned it. The worker interprets the code to determine if the job has been completed successfully. Examples of child processes are SQL*Plus and FNDLOAD.

Deferred Jobs

The first time a job fails, the manager automatically defers the job and assigns a new one to the worker. If the deferred job fails the second time it is run, the manager defers it again only if the total runtime of the job is less than ten minutes. If the deferred job fails a third time (or if the job's total runtime is not less than ten minutes the second time it is run) the job stays at failed status and the worker waits. At this point, you must address the cause of the failure, and then restart the job.

> **Additional Information:** For information about restarting jobs, see Running AD Controller Interactively in this chapter.

The deferred job feature uses the AD_DEFERRED_JOBS table. This table is created when the FND_INSTALL_PROCESSES table is created, and is dropped when the FND_INSTALL_PROCESSES table is dropped.

Determining Number of Workers

The AD utilities provide a default number of workers — two times the number of CPUs on the database server. We recommend you choose a number of workers between two times and four times the number of CPUs. For example, if there are four CPUs on the database server, you should choose from 8 to 16 workers.

The AD utilities calculate a maximum number of workers that your database can support (up to 999). You cannot enter a number of workers greater than the database can support.

Worker Log Files

In addition to the information recorded in the <utility name>.log file, utilities that process jobs in parallel write details about errors to worker log files. The adwork<number>.log files (adwork001.log, adwork002.log...) reside in the $APPL_TOP/admin/<SID>/log directory, where <SID> is the value of the ORACLE_SID or TWO_TASK variable (UNIX), or in %APPL_TOP%\admin\<SID>\log, where <SID> is the value of ORACLE_SID or LOCAL (Windows).

Concurrent requests run by AutoPatch, AutoUpgrade, and AD Administration create their own log files.

> **Additional Information:** See Log and Output Filenames in *Oracle Applications System Administrator's Guide - Configuration.*

Worker Restart Files

Restart files are used to continue processing at the point where it stopped. Each worker may also have a restart file called adwork*xxx*.rf9. These files are stored in $APPL_TOP/admin/<SID>/restart (UNIX) or in %APPL_TOP%\admin\<SID>\restart (Windows). The worker creates the restart file when the manager assigns it a job, and deletes the restart file when it finishes the job.

> **Caution:** Do not modify or delete any manager or worker restart files unless specifically told to do so by Oracle Support Services.

> **Additional Information:** The Troubleshooting chapter in *Oracle Applications Maintenance Procedures* discusses various error situations when running a utility and how to resolve them.

Parallel Support for Data Manipulation Language (DML)

To reduce downtime when creating indexes, the parallel_index_threshold argument for AD utilities is set to a default value of 20,000. This means that if a table contains less than 20,000 blocks, the AD utilities create indexes with parallel workers and serial DML (just as in earlier releases). If a table contains 20,000 blocks or more, indexes are now created with only one worker and parallel DML. You can adjust this threshold value by specifying the parallel_index_threshold argument on the AD utility command line.

Monitoring and Controlling Parallel Processes

AD sessions that use parallel processing may run to completion without user intervention. However, it is often useful to determine how many jobs have been completed or whether processing has stopped for some reason. AD Controller is a

utility that you can use to determine the status of AutoUpgrade, AD Administration, or AutoPatch workers and to control their actions. You can run AD Controller interactively or non-interactively.

> **Additional Information:** See Interactive and Non-interactive Processing on page 1-5.

You choose options that display worker status, restart workers, or issue commands to the manager from the AD Controller main menu.

> **Note:** Run this utility in its own window, not in the same window as AutoUpgrade, AD Administration, or AutoPatch.

Running AD Controller Interactively

Follow these steps to access AD Controller.

1. Log in as applmgr and set the environment as described in Setting the Environment in this chapter.
2. Start AD Controller with the *adctrl* command.

   It prompts you to:

   - Confirm the value of APPL_TOP.
   - Specify an AD Controller log file (the default is adctrl.log). The AD Controller log file is written in the current working directory.
   - Supply the Oracle Application Object Library user name and password.
3. Choose an option from the main menu.

   Once you respond to the prompts, the main menu appears.

```
                AD Controller Menu
    ------------------------------------------------

    1.    Show worker status
    2.    Tell worker to restart a failed job
    3.    Tell worker to quit
    4.    Tell manager that a worker failed its job
    5.    Tell manager that a worker acknowledges quit
    6.    Restart a worker on the current machine
    7.    Exit


Enter your choice [1] :
```

   Type a number to select an option. Press [Return] at any time to return to the AD Controller main menu.

> **Additional Information:** See Troubleshooting in *Oracle Applications Maintenance Procedures* for instructions on using each menu option.

Running AD Controller Non-interactively

You can run AD Controller without user intervention by creating a defaults file, which captures information you supply at the interactive prompts in a file that you can later use to run AD Controller without user intervention. Creating a defaults file and

running AD Controller non-interactively works in much the same way as it does for AD Administration.

> **Additional Information:** See Scheduling Non-interactive Maintenance in *Oracle Applications Maintenance Procedures*.

Like AD Administration, the same defaults file can be used to run different AD Controller commands — a single file can contain all your choices for the different menu options. In order to choose which task the defaults file will run, you add *menu_option*= <menu choice> to the utility start command. This overrides any menu-specific key stroke information stored in the defaults file initially, and allows you to use the defaults file for any of the AD Controller menu items. It also ensures that the menu option you intended for the defaults file is always valid, even it the menu items are renumbered or relocated in subsequent releases

The available options are listed in the following table.

> **Note:** The menu options for running AD Administration are listed in Preparing for Non-interactive Processing in Chapter 4.

Menu Option	Corresponding Menu Choice
ACKNOWLEDGE_QUIT	Tell manager that a worker acknowledges quit
INFORM_FAILURE	Tell manager that a worker failed its job
RESTART_JOB	Tell worker to restart a failed job
SHOW_STATUS	Show worker status
SHUTDOWN_WORKER	Tell worker to quit
START_WORKER	Restart a worker on the current machine

The following is an example of running AD Controller non-interactively to show worker status:

```
$ adctrl interactive=n defaults_file=$APPL_TOP/admin/prod/ctrldefs.txt \
  logfile=adctr.log menu_option=SHOW_STATUS
```

> **Note:** Using any menu option on the command line, except for SHOW_STATUS, requires that you also use the *worker_range*=<*range*> option. See the AD Controller command line help for details.

Distributing Processing Tasks Across Nodes

AD has expanded its existing manager/worker job system used in parallel processing to include *Distributed AD*. This parallel processing feature allows workers in the same AD session to be started on multiple application tier servers to utilize all available resources. Because the AD workers create and update file system objects, as well as database objects, Distributed AD must be used only on systems that are using a shared application tier file system to ensure the files are created in a single, centralized location.

While running either AD Administration or AutoPatch on the primary node, you start an AD Controller session from any of the nodes in the shared application tier file

system environment to perform any standard AD Controller operation, using both local and non-local workers.

> **Additional Information:** See Distributing Processing Tasks in *Oracle Applications Maintenance Procedures*.

OAM Web-based Utilities

When you install or upgrade to this release of Oracle Applications, you gain access to the Oracle Applications Manager (OAM). This web-based management tool allows you to use and access many maintenance utilities that were formerly available only on the command line, and makes it possible to quickly retrieve and display system-specific information in a GUI format. Each utility in OAM is accessed from a main page, which contains links to multiple layers of details that quickly put you in touch with all aspects of your system data. For example, using the Patch Wizard utility, you can access a downloaded list of recommended patches and view the effect on your file system of applying any or all of the patches.

In addition to reporting results based on specific search criteria, many OAM utilities can be used to enter and save changes to your system configuration. For example, using License Manager, you can register products that were not active in your initial installation. Or, with AutoConfig, you can view current configuration parameters and modify the existing values.

> **Additional Information:** See Oracle Applications Manager in *Oracle Applications Concepts*.

Common OAM Operations

The OAM web-based utilities are designed with the same look and feel, making extensive use of common operations such as uniform navigation tools and drill-down menus. For example, all pages present a Help button that opens the OAM Help directory.

In addition, OAM utilities employ a powerful search feature, which displays the search results directly on the page where you initiated the search. No need to review log files or look in a file directory for the report. For example, using the Applied Patches utility, you can perform a simple search for all the patches that have been applied to your system. OAM displays the results at the bottom of Simple Search page.

This section describes several of the most commonly used OAM operations.

Applications Dashboard and Site Map Tabs

The main OAM page is called the Applications Dashboard, which provides a "snapshot" of your Oracle Applications system. From the Applications Dashboard, you can quickly jump to OAM utilities and features in one of two ways: from the Navigate To drop-down list, or by clicking a utility link on the Site Map.

Both the Applications Dashboard and the Site Map page make use of a tab format. Each tab is a link to another part of OAM. Clicking on the tab displays a feature subset. For example, from the Site Map, click on the Maintenance tab to access Patching and Utilities features or Cloning features.

Page Navigation

OAM pages provide several ways to navigate. The Cancel, Back, and Next navigation buttons allow you to move through the page flow. Where appropriate, there are drop-down lists that provide links to related features.

On pages where there are lengthy lists of items, OAM displays a subset of the items. For example, the entire list may contain 75 items, but each page displays only 25. Clicking Next moves the next set of items into view, and clicking Previous moves the prior set into view. You can select items in a subset by clicking an individual item or Select All. To clear selections, click Select None.

Personalized Page Views

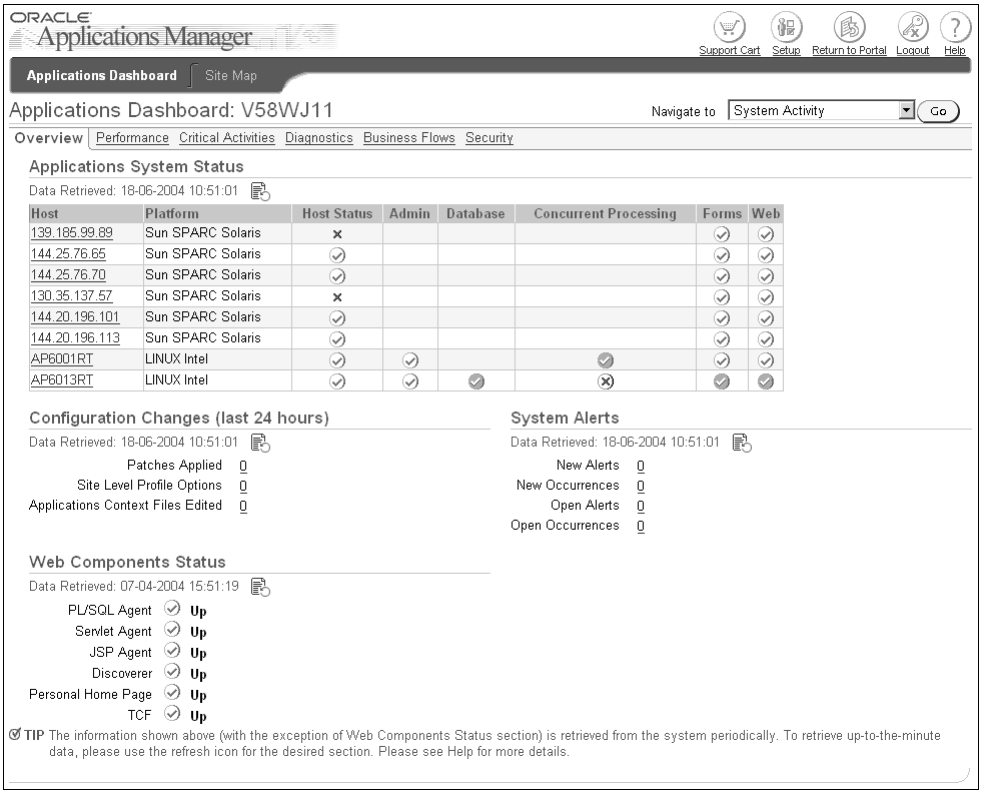
You can alter some of the ways page information is displayed. A blue triangle indicates that topic details can be hidden or displayed. Click it to toggle between the two options. A yellow triangle in a column heading indicates that you can change the sort order of the list from ascending to descending. Click it to toggle between the two options.

OAM Interface

The Oracle Applications Manager Dashboard presents a quick overview of the general status of your system. The Site Map provides access to the utilities and features within the OAM framework.

Reviewing System Status

Regardless of the way you access OAM, the first page it displays is the Applications Dashboard.



The Applications Dashboard serves as an HTML console, where system administrators can view information on general system activity. They can check the status of the database, concurrent managers and other services, concurrent requests, and Oracle Workflow processes, as well as view configuration information, such as initialization parameters and profile options.

Note that there are several regions on the Applications Dashboard: Applications System Status, Configuration Changes, System Alerts, and Web Component Status. Briefly, each region is used as follows:

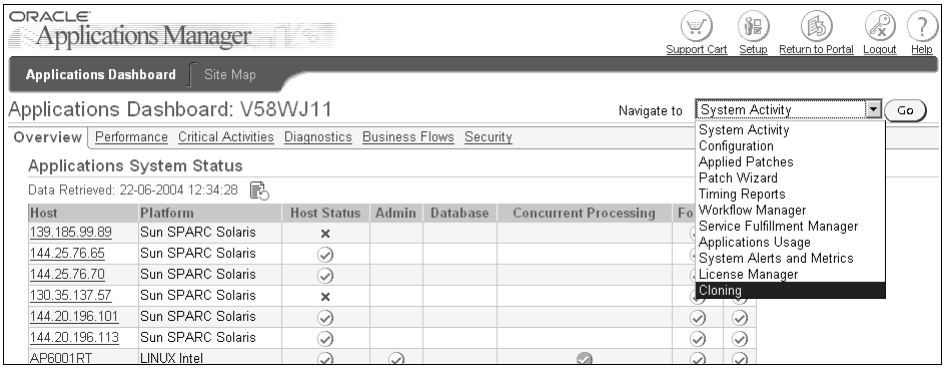
- Applications System Status lists each Host, its Platform, and Status, as well as the status of the services such as Admin and Database.
- Configuration Changes displays the number of patches applied, the number of profile options changed, and the number of context files edited in the last 24 hours.
- System Alerts lists the number of system alerts in several categories including: New Alerts, New Occurrences, Open Alerts, and Open Occurrences.
- Web Components Status lists the status of the web components, including PL/SQL Agent, Servlet Agent, JSP Agent, JTF, and others.

As the gateway to the Oracle Applications Manager, the Applications Dashboard is used in various ways. Many of the functions are beyond the scope of this book, and so are documented elsewhere.

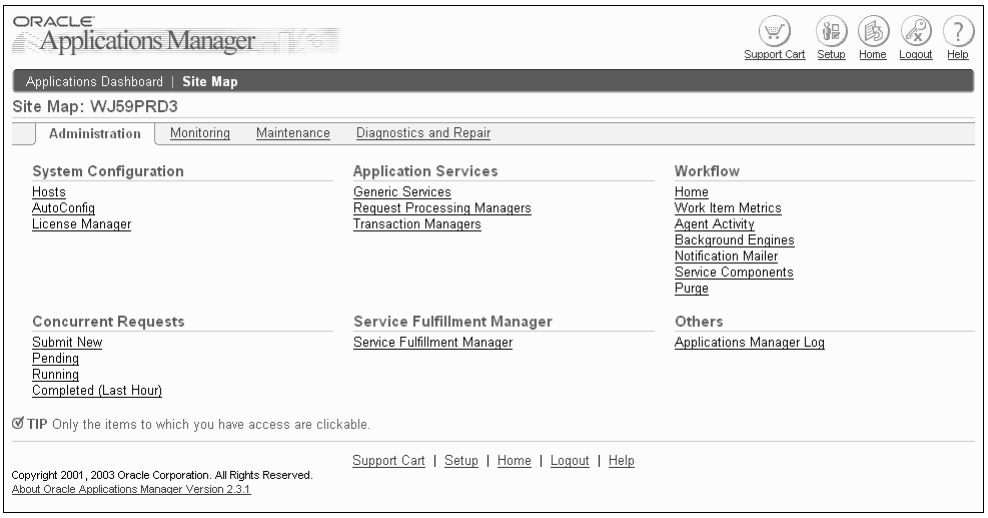
> **Additional Information:** See Oracle Applications Manager in *Oracle Applications Concepts*. See also OAM Help for more details.

Accessing Maintenance Utilities

The Applications Dashboard is also the gateway to the web-based maintenance utilities discussed in this book. Some of the utilities, like License Manager and Patch Wizard, are available directly from the drop-down list.



Just open the list, make a selection, and click Go. For a more complete list of all the utilities and features included in OAM, click the Site Map tab.



The Site Map page displays tabs for Administration, Monitoring, Maintenance, and Diagnostics and Repair. On individual tabs, there are links to utilities or functions under general groups. For example, on this page there are headings for System Configuration, Application Services, Workflow, and so on.

To open the main page for a utility, find it under one of the headings and click the link. For example, to register products or languages in your system after the initial installation or upgrade, use the License Manager. It is located under the System Configuration heading.

Click any of the other tabs to access other functions. All instructions for accessing OAM web-based utilities in this book, and in *Oracle Applications Maintenance Procedures*, start from the Site Map.

Accessing OAM

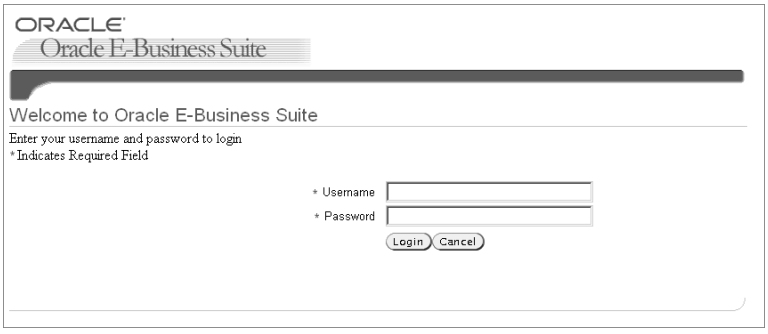
You can access OAM in one of two ways: by entering its URL directly, or by accessing it from the Rapid Install Portal.

Using the URL:

You access OAM by typing the following URL in your browser:

```
http:.// <HTTP hostname>.<domain>:<HTTP port>/servlets/weboam/oam/oamLogin
```

The Welcome page appears.



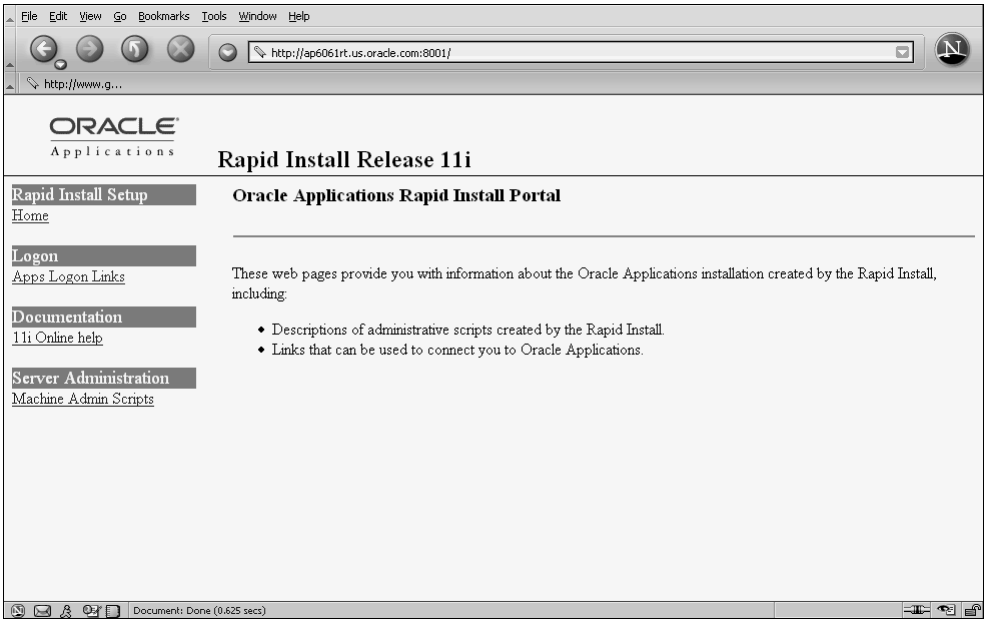
Enter your username and password, and click Login. The OAM Applications Dashboard appears.

Using the Rapid Install Portal:

You can also access the OAM Dashboard from the Rapid Install Portal page. Type the following URL in your browser:

```
http://<HTTP hostname>.<domain>:<HTTP port>
```

The Rapid Install Portal page appears.



Click Apps Logon Links to see the logon links. Click Oracle Applications Manager. The Applications Dashboard appears.

2

Installation and Upgrade

This chapter describes the utilities you use to install a new Oracle Applications system and to upgrade an existing system to a new release version. It contains these sections:

- About Installations and Upgrades
- Rapid Install
- AutoUpgrade

About Installations and Upgrades

This section briefly describes the installation and upgrade process, and introduces Rapid Install and AutoUpgrade — the command line utilities you use to install a new Oracle Applications system or upgrade Oracle Applications products in an existing system to a new release level. Subsequent sections in this chapter contain details about using these utilities.

Installing a New System

A new installation of Oracle Applications is managed by Rapid Install. It sets up a fully configured system, including a database, a complete set of Oracle Applications E-Business Suite products, the latest certified technology stack, all patches, minipacks, family packs, and other updates that are available at the time of the software release.

All *products*, regardless of their licensed status, are installed. During the installation, you will have an opportunity to flag the products you have licensed to register them as active in your system.

The *technology stack* consists of the components that are required to run the new system — those specific to both the database tier and the application tier. For example, a new installation includes a fresh database (and the associated Oracle homes) certified to work with the version of Oracle Applications, as well as the latest application tier components like Oracle *i*AS server and Oracle Developer.

> **Additional Information:** The installation process and the Rapid Install wizard are described in detail in *Installing Oracle Applications*.

Upgrading an Existing System

A complete system upgrade updates the technology stack and upgrades the installed Oracle Applications products from one release level to a new one. It does not install a fresh database, but instead installs the technology stack components you need to upgrade your existing database to the latest certified release level.

AutoUpgrade manages the product upgrade process. It upgrades the installed products from the existing release level to the base version of a new release level. In addition, it marks the dividing point between the two parts of a complete system upgrade: pre-upgrade processing and post-upgrade processing. The *pre-upgrade* process prepares your system and products for running AutoUpgrade. The *post-upgrade* process brings your system and products up to the latest maintenance level release.

The following utilities are also required to complete a system upgrade:

- Rapid Install

  Employs a specialized upgrade screen flow to collect configuration parameters for the upgraded system.

- AutoPatch

  Used at various times during the upgrade process to apply product patches, and during the post-upgrade process to bring your Oracle Applications system up to the full maintenance release level.

> **Additional Information:** The upgrade process is described in detail in *Upgrading Oracle Applications*. AutoPatch functionality extends beyond the upgrade process. It is described in detail in Chapter 5.

Rapid Install

Rapid Install is used to install a new Oracle Applications system. Using the configuration parameters you enter on the Rapid Install screens, it creates the Oracle Applications file system, installs a database (for new installations), registers products as being licensed and active in your system, manages NLS system requirements, configures various port connections, and creates and runs the start and stop scripts for the database and its listeners.

It also forms an integral part of a complete system upgrade, using configuration parameters similar to those you enter for a new installation, but tailored to meet the requirements for upgrade processing. It stores those parameters, and then uses them both during the pre-upgrade process and the post-upgrade process.

Regardless of whether you run Rapid Install to create a new system or to upgrade an existing one, the configuration values you enter are stored in a context file, which you can use later to modify your system configuration.

> **Additional Information:** See Managing Configuration *Parameters in Oracle Applications Maintenance Procedures*.

New Installations

Rapid Install offers Custom and Express installation options. Both give you the choice of installing a *fresh* database — one that is fully configured but contains no transaction data — or a *Vision Demo database* — one that contains transaction data for a fictitious company, which you can use for training or demonstration purposes.

In a custom installation, you define the configuration, and choose whether the installed system will run on a single node, or on multiple nodes. In an express installation, you set up a single-user, single-node system using a few basic, user-defined configuration parameters. Rapid Install supplies the remaining parameters based on system default values.

Rapid Install can also be directed to set up a multi-node system that uses a shared application tier file system or automatically employs load balancing — directing forms requests to the least loaded forms server.

Upgrades

Rapid Install also plays a vital part in the upgrade process. You run it twice — first to lay down the file system and create a working environment before you upgrade your products, and a second time to configure and start server process for the upgraded database.

The Rapid Install Interface

Rapid Install starts from the command line. However, its processing operations do not resemble the ones described in AD Command Line Utilities in Chapter 1. Instead, it displays a Java-based screen flow called the Rapid Install wizard, which presents a series of screens that step you through the process of collecting configuration parameter values.

You indicate the screen flow you want to use, for example, to set up a new installation or upgrade an existing one, and the wizard displays default configuration values for that process. Using a combination of text input boxes and drop-down lists, you either accept the defaults or enter new values.

Running Rapid Install

This section describes the basic start operation, and provides examples of some of the Rapid Install screens. For a complete description of Rapid Install, including examples of all the screens in the Rapid Install wizard, see *Installing Oracle Applications*.

1. Install Java Development Kit (JDK).

   Rapid Install installs JDK automatically on Solaris, Linux, and Windows operating systems. If you are running on another platform, see *Oracle Applications Installation Update Notes* for platform-specific information.

2. Set up a stage directory.

   To shorten installation time, and eliminate the need for user access to the installation media during the installation, start Rapid Install from a stage area. To do this, copy the media to a file system and run the installation from there. See Set Up Stage Area in *Installing Oracle Applications*.

3. Start the Rapid Install wizard.

   Use the following commands to start the wizard.

   UNIX:

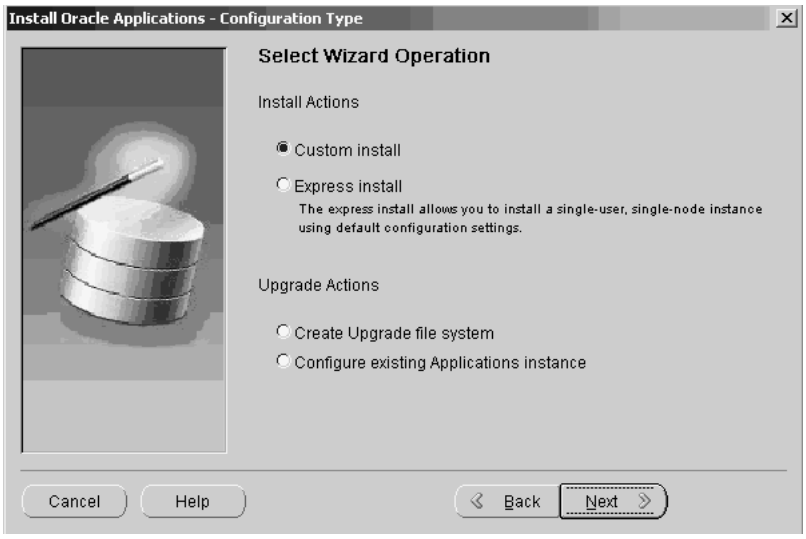
   ```
   $ cd /u01/Stage11i/startCD/Disk1/rapidwiz
   $ rapidwiz
   ```

   Windows:

   ```
   C:\>f:
   F:\>cd Stage11i\startCD\Disk1\rapidwiz
   F:\Stage11i\startCD\Disk1\rapidwiz> rapidwiz.cmd
   ```

4. Choose the type of installation.

From the Installation Type screen, you choose the appropriate screen flow. For example, to install a new system, using custom configuration parameters, click Custom install.



When you make a selection on this screen, and click Next, the wizard continues with the appropriate screen flow.

5. Complete the wizard screens.

   The remaining wizard screens prompt for the system configuration parameters necessary to complete the installation or upgrade. Once you complete the wizard, Rapid Install verifies that all parameters necessary to create working environment are present, and begins to set up your system.

AutoUpgrade

AutoUpgrade is used to upgrade the products installed in your system from your existing release level to the base version of a new release level. You start this utility from the command line, respond to a series of prompts, and then select the appropriate actions using the options on the AutoUpgrade main menu.

Each main menu option represents a step in the process of running AutoUpgrade. You can choose database parameters and overall upgrade tasks, and then run the selected tasks to initiate the product upgrade. Where necessary, a task from the main menu displays an AutoUpgrade screen where you can fine-tune the actions AutoUpgrade will take during processing.

Prompts

Chapter 1 describes the command line prompts commonly displayed by most AD utilities. However, AutoUpgrade also prompts you for additional information that is unique to the upgrade process. While most of the AutoUpgrade prompts are self-explanatory, the extra detail in this section may be helpful.

Note: Many AutoUpgrade prompts verify information that was set up when you ran Rapid Install during the pre-upgrade process.

AutoUpgrade log file name

The AutoUpgrade log file is where the AutoUpgrade manager stores information about actions taken during the upgrade. Enter a new name or accept the default (adaimgr.log). The file is located in $APPL_TOP/admin/<SID>/log, where <SID> is the value of the ORACLE_SID or TWO_TASK variable (UNIX), or in %APPL_TOP%\admin\<SID>\log, where <SID> is the value of the ORACLE_SID or LOCAL variable (Windows).

Software compatibility

```
Are you certain you are running a certified release combination [No] ?
```

This prompt asks you to confirm that your existing system is made up of Oracle Applications products and a database that are certified by Oracle to be compatible. For example, an Oracle8*i* (8.1.7.4) database is certified to work only with certain releases of Oracle Applications products. You can verify compatibility on the Certify web page on Oracle*MetaLink* (Doc ID: 119139.1).

Applications system name

```
Please enter the name of the Oracle Applications System that this APPL_TOP belongs
to.

The Applications System name must be unique across all Oracle
Applications Systems at your site, must be from 1 to 30 characters
long, may only contain alphanumeric and underscore characters,
and must start with a letter.

Sample Applications System names are: "prod", "test", and "demo" and
Development_2.

Applications System Name [prod] :
```

You set the Applications system name when you ran Rapid Install during the pre-upgrade process (default <SID>). This prompt asks you to verify that name. If you want to change the name, you must do so after the upgrade is complete.

Files installed in the APPL_TOP

```
NOTE: If you do not have or choose not to have certain types of files installed
in this APPL_TOP, you may not be able to perform certain tasks.

Example 1: If you don't have files used for installing or upgrading
the database installed in this area, you cannot install or upgrade
the database from this APPL_TOP.

Example 2: If you don't have forms files installed in this area, you cannot
generate them or run them from this APPL_TOP.

Example 3: If you don't have concurrent program files installed in this area,
you cannot relink concurrent programs or generate reports from this APPL_TOP.

Do you currently have or want to install files used for installing or upgrading
the database in this APPL_TOP [Yes] ? Yes *

Do you currently have or want to install Java and HTML files for HTML-based
functionality in this APPL_TOP [Yes] ? Yes *

Do you currently have or want to install Oracle Applications forms files
in this APPL_TOP [Yes] ? Yes *
```

```
Do you currently have or want to install concurrent program files
in this APPL_TOP [Yes] ? Yes *
```

The server configuration information was set by Rapid Install during the pre-upgrade process, based on configuration values you entered in the wizard. At that time, you determined what types of files (such as Java files, HTML files, forms files, and concurrent program files) were unloaded on a given node. The defaults in this prompt pertain to this server configuration. Do not change any of the defaults at this time (unless instructed to do so by Oracle Support Services).

APPL_TOP name

```
Please enter the name Oracle Applications will use to identify the APPL_TOP.

The APPL_TOP name you select must be unique within an Oracle Applications
System, must be from 1 to 30 characters long, may only contain.
alphanumeric and underscore characters, and must start with a letter.

APPL_TOP Name [prod_all] :

Please enter the name Oracle Applications will use to identify this APPL_TOP.
```

Like the Applications system name, the APPL_TOP name was also created when you ran Rapid Install in the pre-upgrade process. It has been stored as a configuration parameter, and is used by all the other AD utilities. We recommend that you do not change the APPL_TOP name.

Identify Organization Type

AutoUpgrade automatically determines the type of organization your products are associated with: commercial (for profit), government (public sector), education, or not-for-profit.

Commercial product installations

If your existing installation is using commercial products, AutoUpgrade asks whether you want to continue with those products, or convert them to another type.

```
1) Continue to use Oracle Applications for Commercial or for-profit use.
2) Convert Oracle Applications to government, education or not-for-profit use.
```

> **Note:** AutoUpgrade displays this prompt only if your existing system was set up as a commercial installation.

Choose Option 1 to indicate that your organization will continue as commercial (for profit) or as a not-for-profit organization. Choose Option 2 to convert existing commercial products to either government (public sector), education, or not-for-profit.

> **Caution:** Products installed as not-for-profit cannot be converted to commercial products.

Government (public sector) product installations

In addition to terminology changes to messages and other components, a public sector installation completes the following actions:

- Creates account types for Budgetary Dr and Budgetary Cr.
- Creates new profiles for INDUSTRY (Industry), ATTRIBUTE_REPORTING (FSG: Enable Attribute Reporting), ACCOUNT_CATEGORY_BALANCING (GL: Enforce Account Category Balancing), CREATE_BUDGETARY_ENCUMBRANCES (GL: Create Encumbrances for Budgetary Accounting), and USSGL_OPTION (Transaction Code).
- Enables concurrent programs for GLGHIST (Program - Reporting Attribute Historical), GLGPREP (Program - Reporting Attribute Preparation), GLGEOT (Other - Open Encumbrance Balance), GLGFUN (Budget - Funds Available Analysis), GLXRLTCL (Other - Transaction Code Listing), GLGENCRE (Encumbrance - Encumbrance Account Details Report), and GLGDOCDE (Encumbrance - Encumbrance Document Details Report).

A public sector installation is required if you use the USSGL model of budgetary control using transaction codes and the budgetary account types, or if you use FSG reporting with the accounting flexfield segment attributes.

Use of encumbrance accounting, budgetary control using encumbrances, or the Multi-Fund Accounts Receivable feature in Oracle Public Sector Advanced Features does not require a public sector installation.

Error Messages and Warnings

Error messages and warnings are recorded in the AutoUpgrade log file. Some require corrective action, and some do not. This section lists types of errors and warnings that are acceptable — you can ignore them and continue the AutoUpgrade session by restarting the failed job with AD Controller. Contact Oracle Support Services if the worker encounters the same error again.

> **Additional Information:** See Running AD Controller Interactively in Chapter 1. See also Troubleshooting in *Oracle Applications Maintenance Procedures*.

Oracle Database Error Messages

The following Oracle database error messages (ORA errors) indicate acceptable conditions and do not require any corrective action:

```
ORA-00942: table or view does not exist
ORA-00955: name is already used by an existing object
ORA-01418: specified index does not exist
ORA-01430: column being added already exists in table
ORA-01434: private synonym to be dropped does not exist
ORA-01442: column to be modified to NOT NULL is already NOT NULL
ORA-01451: column to be modified to NULL cannot be modified to NULL
ORA-04043: object <object name> does not exist
ORA-04080: trigger '<trigger name>' does not exist
```

This action required for some ORA error messages depends on your system configuration or where in the upgrade processing the error occurred. For example, the "ORA-0060: deadlock detected while waiting for resource" message typically indicates that another process has accessed the resources need for the upgrade process. When the resource is free, the upgrade can continue — no other action is required.

It's a good idea to get more information about ORA error messages that you don't understand. To see a definition of a message, type the following on the command line:

```
$ oerr <message_number>
```

Additional Information: See *Database Error Messages* for complete information.

If you consistently receive ORA-1555 errors, the problem may be due to insufficient rollback space. Try increasing the available rollback space before restarting the workers.

Additional Information: See Managing Rollback Segments in *Oracle Administrator's Guide.*

Database Object Differences

Several warnings may be reported when AutoUpgrade compares database objects in the upgrade release version to those from the previous release version:

```
Warning: The missing column is NOT NULL
Warning: The existing index is UNIQUE and the new index is nonunique.
Warning: NO default value can be applied.
```

AutoUpgrade may also indicate that extra database objects exist, or that there are differences in the constraints of columns. Such warnings and differences are usually not a problem. The differences can be due to database customizations, or because obsolete objects or columns were not dropped by Oracle Applications during the upgrade. If you have custom database objects that rely on these Applications objects, review these warnings after the upgrade to determine whether you need to modify your customizations.

Import (IMP) Messages

When AutoUpgrade runs the Oracle Import utility (IMP) to migrate your existing database files to the upgraded database, it may generate the following warnings. Typically, these warnings do not require corrective action.

```
IMP-00041: Warning: object created with compilation warnings
Warning: the objects were exported by SYSTEM, not by you
```

DataMerge Error Messages

DataMerge (addmimp) runs during an upgrade to import seed data needed by Oracle Applications. A message like the following indicates a problem with a DataMerge temporary file:

```
addmimp: <function name> : error : <error with temporary file>
```

Here, <function name> is the name of a DataMerge function, such as *dmmactin* or *dmmactwrt*. The error descriptions vary, but always specify a problem with a temporary file.

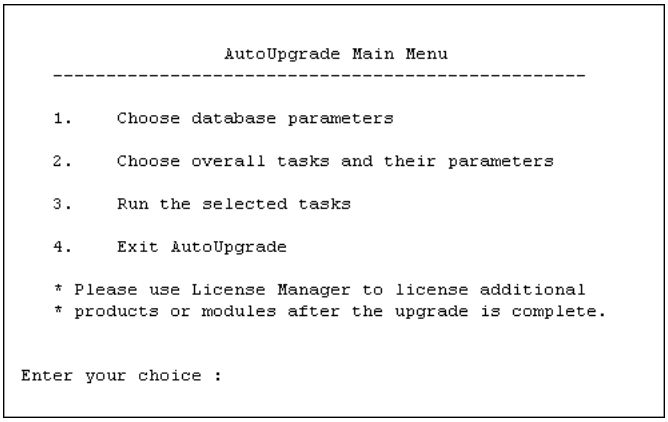
To fix this problem, try restarting AutoUpgrade or the AutoUpgrade worker immediately after you receive this error. Persistent DataMerge errors may indicate a problem with the APPLTMP variable, a lack of available space in the temporary directory, insufficient rollback space, or incorrect access privileges on the directory.

The AutoUpgrade Interface

You start AutoUpgrade from the command line. However, all upgrade processes are initiated from the AutoUpgrade Main Menu. This section describes some of the common features of the AutoUpgrade interface.

Main Menu

The AutoUpgrade Main Menu provides access to the tasks required to set up and run AutoUpgrade.



Options 1 and 2 set up parameters and select the upgrade tasks you want AutoUpgrade to perform. Option 3 runs the selected tasks using the parameters you specified. If you choose Option 4 to exit AutoUpgrade, be sure to save the restart files, when prompted, so you can run AutoUpgrade again without having to re-enter the information.

Option Numbers

Items listed on the screens, including menu options, products, and tasks, are numbered. To select a list item, type the number at the prompt at the bottom of the screen.

Navigation

If it is necessary to navigate through a list of items that is too lengthy to be displayed on a single screen, you can use the following choices to move through the list.

- Up (U) moves the previous group of items into view.
- Down (D) moves the next group of items into view.
- Top (T) moves to the top of the item list.
- Bottom (B) moves to the bottom of the item list.
- [Return] sends you back to the Main Menu.

Running AutoUpgrade

Complete the steps in this section to display the AutoUpgrade Main Menu and run the upgrade tasks.

> **Note:** Although AutoUpgrade can be used with non-graphic terminal devices, we recommend using a window-based terminal so you can monitor and control upgrade processes in other windows while AutoUpgrade is running.

Caution: Use AD Controller to shut down workers if you need to stop an upgrade process selected from the main screen. If an error occurs while you are upgrading the database, you can correct the error without stopping AutoUpgrade.

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Start AutoUpgrade

From any directory, start AutoUpgrade with one of the following commands.

If you are converting to the new Oracle Applications Tablespace Model (OATM):

```
$ adaimgr
```

If you are using the OFA-compliant (old) tablespace model:

```
$ adaimgr consolidated_tablespace=n
```

The utility starts and displays the first prompt.

Step 3 Respond to prompts

Complete the information in the AutoUpgrade prompts. The basic AD command line prompts are described in Chapter 1. Prompts that are unique to AutoUpgrade are described in this chapter on page 2-4.

When you complete the prompts, the Main Menu appears.

Step 4 Choose upgrade tasks

On the Main Menu, type 1 and press [Return]. The Choose Database Parameters screen appears.

Step 5 Choose database parameters

This screen presents a numbered list of installed products in the Product Name column along with parameters that you can specify for these products, identified by a row of letters across the top of the screen (O, S, M, I, D).

Note: If you migrate to the Oracle Applications Tablespace Model, the product tablespaces default to precreated tablespaces and the Choose database parameters option does not appear on the AutoUpgrade main menu. See Set Up Tablespaces (Database Upgrade Tasks) in Chapter 2 of *Upgrading Oracle Applications*.

```
                AutoUpgrade - Choose database parameters

                                 - O -    - S -   --- M ---   --- I ---   --- D ---
  Product              Action    ORACLE   Sizing  Main        Index       Default
# Name                   |      User ID  Factor  Tablespace  Tablespace  Tablespace
-- ---------------------  -  ------- ------ ---------- ---------- ----------
1 Application Object Lib     APPLSYS     100  APPLSYSD   APPLSYSX   APPLSYSD
2 Application Utilities      APPLSYS     100  APPLSYSD   APPLSYSX   APPLSYSD
3 Applications DBA           APPLSYS     100  APPLSYSD   APPLSYSX   APPLSYSD
4 Oracle Alert               ALR         100  ALRD       ALRX       ALRD
5 Global Accounting Engi     AX          100  AXD        AXX        AXD
6 Oracle Common Modules-     AK          100  AKD        AKX        AKD
7 Oracle Common Accounti     XLA         100  XLAD       XLAX       XLAD
8 Oracle General Ledger      GL          100  GLD        GLX        GLD

     There are 208 Oracle Applications.  Enter U/D to scroll up/down.


<Product #><Letter> - To change a database parameter for a product;
                      INCLUDE the LETTER ABOVE the COLUMN you want to change
  U / D / T / B     - Press up/down/top/bottom to see other products
    [Return]        - To return to the AutoUpgrade Main Menu

Enter your choice (for example, 1M) : _
```

Note the following restrictions and guidelines for entering parameters.

- We recommend that you accept the default schema names for all products unless they conflict with other schema names you already have.
- For security reasons, we recommend you change all passwords and use no default passwords.
- You cannot change the usernames and passwords for the following products. They use the Oracle Application Library username (APPLSYS), which is determined by AutoUpgrade.
  - Oracle Application Object Library (AOL)
  - Oracle Applications DBA (AD)
  - Oracle Applications Utilities (AU)
  - Oracle Applications Shared Technology (SHT)
- You can change the password for Oracle Application Object Library after you run AutoUpgrade.

  Additional Information: See Oracle Applications Schema Password Change Utility in *Oracle Applications System Administrator's Guide - Configuration*.

- Oracle Human Resources, Oracle Payroll, FastFormula, and DateTrack must be installed under the same schema. If you change the schema for one of these products, AutoUpgrade automatically changes the schema for the others.

The Action column is for information only. It indicates whether AutoUpgrade will upgrade that product. All registered — licensed and marked as active — products are marked U, and will be upgraded. A blank entry means that the product is not registered as active in your system, and will not be upgraded by AutoUpgrade.

> **Additional Information:** You can register additional products after the upgrade is complete. See Registering Applications Products in *Oracle Applications Maintenance Procedures*.

The database parameters columns indicate which parameters can be modified for each product. They are displayed across the top of the screen. Above each column is a letter that you use to indicate that parameter in the screen prompt.

Oracle User ID (O)

The Oracle User ID (username/password) that owns each product's database objects. You cannot change the User ID for products that are currently installed and in use.

To change any of the other Oracle User IDs, type <product number>O at the prompt at the bottom of the page. AutoUpgrade asks for the Oracle username. Accept the default or enter a new username (up to 30 characters). To change only the password, press [Return] when AutoUpgrade prompts for the username, and enter the new password at the next prompt.

Sizing Factor (S)

This is the sizing factor that AutoUpgrade applies to *new* product tables and indexes and to new database objects added during the upgrade. It does not affect objects already installed in the database.

Products that are not registered (as being licensed and active) are installed with the sizing factor shown as the default on this screen. We recommend that you leave the sizing factor at the default (100), so you can license new products and register them for immediate use. Once registered, set the product's table and index next extent based on use.

To change a sizing factor, type <product number>S and enter the new sizing factor at the prompt.

Tablespaces (M, I, D)

The Main Tablespace (M) and Index Tablespace (I) are the Oracle server tablespaces in which AutoUpgrade places product tables and indexes. The Default Tablespace (D) is used for operations that do not specify a tablespace, and defaults to the main tablespace for that product. The default tablespace names follow a standard naming convention using the product's abbreviation followed by a D for data or an X for indexes. For example, GLD and GLX for the GL product.

AutoUpgrade verifies tablespace names when you enter them, and when you return to the Main Menu. If you specify a new main or index tablespace, AutoUpgrade places only new database objects in those tablespaces. It does not move existing tables or indexes to the new tablespaces.

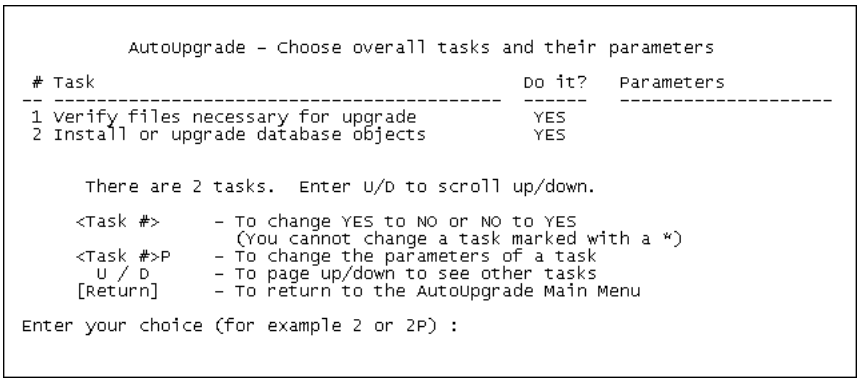
To change the tablespace for a product, enter the product number explicitly. For example, <product number>M, <product number>I, or <product number>D. You cannot change the Main tablespaces or the Index tablespaces for all installed products at the prompt — that means *AM* and *AI* are not valid entries.

Note: You can change a database parameter for all products by entering A (instead of a product number) along with a parameter letter.

Complete the information on this screen, and press [Enter] to return to the Main Menu.

Step 6 Choose overall tasks

From the Main Menu, enter 2 at the prompt and press [Return]. The Choose Overall Tasks and their Parameters screen appears.



The Task column displays the tasks AutoUpgrade will perform during the upgrade processing. For each task, the Do It? column indicates the default action. You can toggle the action from Yes to No by typing the number of the task at the prompt at the bottom of the screen.

Verify files necessary for upgrade (Option 1)

When you choose this option, AutoUpgrade verifies that all files necessary for the upgrade are present. You can run this task independently as a pre-upgrade step to identify missing files before you proceed with the upgrade process.

> **Additional Information:** See Overview of an Upgrade in *Upgrading Oracle Applications*.

Install or upgrade database objects (Option 2)

Choosing this option causes AutoUpgrade to verify files, upgrade database objects for existing product groups, and install new database objects.

To return to the Main Menu, press [Return].

Step 7 Run selected tasks

From the Main Menu, enter 3 (Run the Selected Tasks) at the screen prompt to start the upgrade. AutoUpgrade performs the tasks you marked with Yes on the Choose Overall Tasks and their Parameters screen.

During the upgrade process, AutoUpgrade performs the following tasks:

- Prompts for the number of workers

  Enter the number of workers you want to use for processing the upgrade tasks.

  > **Additional Information:** See Using Parallel Processing in Chapter 1 for more information on parallel processing.

- Verifies files and prompts for output log file.

  As AutoUpgrade verifies files, it asks for the name of the log file in which the output from these tasks should be stored:

  ```
  Please enter the filename you wish to use or press [RETURN] to accept the
  default filename [adiuvf.lst]:
  ```

  The log file (adiuvf.lst) is stored in $APPL_TOP/admin/<SID>/out (UNIX), or %APPL_TOP\admin\<SID>\out (Windows). You can accept the default name or enter a new one at this prompt.

If AutoUpgrade finds that any files are missing, it displays a failure message and stops. It lists the missing files in the adaimgr.log file. Review the missing files, correct the problems, and restart AutoUpgrade.

- Validates pre-upgrade steps

  AutoUpgrade verifies that all the required pre-upgrade steps have been run and records that information in the log file.

- Installs and upgrades database objects.

  All previously licensed (registered as active) products, and the database objects for all products in the new release, are upgraded.

When the processing is complete, AutoUpgrade returns you to the Main Menu.

Step 8 Exit AutoUpgrade

When the Main Menu appears, enter 4 (Exit AutoUpgrade) at the screen prompt. You can also choose to exit AutoUpgrade any time you have access to the Main Menu. For example, you can choose to exit after you have completed the Database Parameters screen information.

Restarting AutoUpgrade

Any time you choose Exit AutoUpgrade from the Main Menu, the utility saves the actions it has taken up to that point in restart files. When you restart AutoUpgrade, choose Yes at the following prompt to restart from the point where you exited:

```
Your previous AutoUpgrade session did not run to completion.
Do you wish to continue with your previous AutoUpgrade Session [Yes]?
```

Additional Information: See Restart Files in Chapter 1.

Correcting Worker Errors

When a worker fails its job, you can attempt to fix the failed job while the other workers and the manager are still running. Use the worker log files to determine the problem and restart the worker while the manager is running.

Additional Information: See Troubleshooting in the *Oracle Applications Maintenance Procedures* for information on handling a failed worker. See also Running AD Controller Interactively in Chapter 1.

3

Configuration

During an installation or upgrade, your system is set up and configured based on the values you specify as a part of those processes. At various times after an installation or upgrade, you may need to reconfigure your system. Oracle Applications employs several utilities to aid you with this task. This chapter contains information about the following utilities:

- About System Configurations
- AutoConfig
- License Manager
- AD Splicer
- File Character Set Converter

About System Configurations

During a new installation or an existing system upgrade, you define the configuration of your system by registering products and languages and supplying information such as database name, top-level directories and mount points, products and country-specific functionalities, NLS settings, and global network settings. These values are propagated to the individual system configuration files, and also stored in a central repository file called a context.

Your system configuration can be changed as needed by using one of several utilities designed to report on and manage the configuration information. Some of these utilities are accessed from the command line and some are web-based.

Web-based Configuration Utilities

The following utilities are web-based. You access them through the Oracle Applications Manager (OAM).

AutoConfig

System configuration parameters are stored and managed by AutoConfig. Rapid Install uses this utility to set up the system configuration during an installation or upgrade. However, you can also access AutoConfig independently to view and edit the individual configuration parameters that define your system.

License Manager

Products, country-specific functionalities (localized products), and languages that you license or begin to use after the initial installation must be registered as active in order to be included in various system maintenance tasks. Using License Manager, you can

create reports about currently registered products and register additional products, country-specific functionalities (localized products).

Command Line Configuration Utilities

These AD utilities are run from the command line.

AD Splicer

Splicing refers to the process of adding a product that was not included in a base release to the products in an existing system. AD Splicer modifies the APPL_TOP and database so that AutoPatch and AD Administration recognize the product as valid.

File Character Set Converter

This utility converts the character set of individual files — those not included in processing performed by AD Administration, AutoPatch, or Rapid Install — to the character set used in your system.

AutoConfig

Rapid Install uses AutoConfig during an installation or upgrade to create two XML-based context files, one for the application tier and one for the database tier. It stores the configuration parameters for your system in these files.

> **Additional Information:** See AutoConfig in *Oracle Applications Concepts.*

After your system is set up, AutoConfig stores these context files in a central location in the Oracle Applications Manager. There, you can select a context file, view existing parameters and modify them, and compare existing values with previous ones.

When you modify the values in a context file, OAM stores them in the OAM schema and automatically updates (synchronizes) the context files on the Applications file system. Then, you run an AutoConfig command line script to propagate the new values to the configuration files in both the APPL_TOP technology stack and the ORACLE_HOME technology stack.

> **Additional Information:** See Propagating Configuration Changes in *Oracle Applications Maintenance Procedures.*

Patched Files

When you apply a patch that contains a new version of a file maintained by AutoConfig, AutoPatch runs the AutoConfig script automatically to propagate new values to the configuration files, so you need not run the script as a separate step.

The AutoConfig Interface

AutoConfig is a web-based utility in Oracle Applications Manager. It consists of these pages: the AutoConfig home page, the Edit Parameters page, the Show History page, and the Differences from Current Configuration page. Since this is a web-based utility accessed through OAM, all the pages share a uniform look and feel.

> **Additional Information:** See OAM Interface in Chapter 1.

Home page

Lists the context files and indicates whether it is for the application tier or the database tier. It contains the links to the Edit Parameters page and the Show History page.

Edit Parameters page

Lists the existing parameters in the context file you selected on the home page. The parameters are grouped by category: Global, System, Local, Environment, Processes, and Custom. Use this page to select and edit the value of individual parameters.

Show History page

Lists other context files stored in the database (used in previous configurations). Use this page to select an older context file to compare with the existing one.

Differences from Current Configuration

Lists the differences in the parameters between an older configuration and the existing configuration. Use this page to select individual parameters that you want to use to replace existing ones.

> **Additional Information:** Click the OAM help button and navigate to the AutoConfig topic for more details about these pages.

Accessing AutoConfig

To access AutoConfig, log in to Oracle Applications Manager and choose AutoConfig from the Site Map.

1. Log in to Oracle Applications Manager.

   Follow the instructions in Accessing OAM in Chapter 1. From the Applications Dashboard, click the Site Map tab.

2. Go to the AutoConfig home page.

   From the Site Map, AutoConfig is on the Administration tab under the System Configuration heading. Click the AutoConfig link to go to the home page.

3. View or edit system parameters.

   From the AutoConfig home page, you can view context files or select a file and edit the parameters or show a history of other context files.

4. Close AutoConfig.

   When you have finished editing parameters, click the Site Map tab to access other OAM utilities, or click Logout to close OAM.

   Additional Information: See Managing Configuration Parameters in *Oracle Applications Maintenance Procedures* for detailed instructions for using AutoConfig.

AutoConfig Scripts

After updating the context files, you propagate the changes to the configuration files by running the AutoConfig command line script. The primary AutoConfig script is adautocfg.sh (UNIX) or adautocfg.cmd (Windows) and is located in the COMMON_TOP/admin/scripts/<CONTEXT_NAME> directory. This script calls adconfig.sh (UNIX) or adconfig.cmd (Windows), which then runs the adconfig.pl script to perform the actual configuration tasks. In some cases where environment variables must be entered manually, you run the adconfig.pl script directly.

Additional Information: See Generating Updated Configuration Files in *Oracle Applications Maintenance Procedures* and Merging Existing APPL_TOPs in *Oracle Applications Maintenance Procedures*.

AutoConfig Test Mode

Running the AutoConfig script to propagate changes overwrites the existing configuration files. This process may overwrite any customizations you made to the configuration files. To prevent any unwanted changes, use the test mode of AutoConfig to determine the impact of running the AutoConfig script without actually making the updates.

The AutoConfig test mode script, adchkcfg.sh (UNIX) or adchkcfg.cmd (Windows), is located in <AD_TOP>/bin on the application tier and in <RDBMS ORACLE_HOME>/appsutil/bin on the database tier.

Note: The AutoConfig test mode script runs automatically when you run AutoPatch in test mode. See AutoPatch Modes in Chapter 5

To run the script on the application tier:

UNIX:

The script prompts for the location of the Applications context file and the APPS password.

```
$ <AD_TOP>/bin/adchkcfg.sh
```

Windows:

The script prompts for the location of the Applications context file and the APPS password.

```
C:\> <AD_TOP>\bin\adchkcfg.cmd
```

To run the script on the database tier:

UNIX:

The script prompts for the location of the database context file and the APPS password.

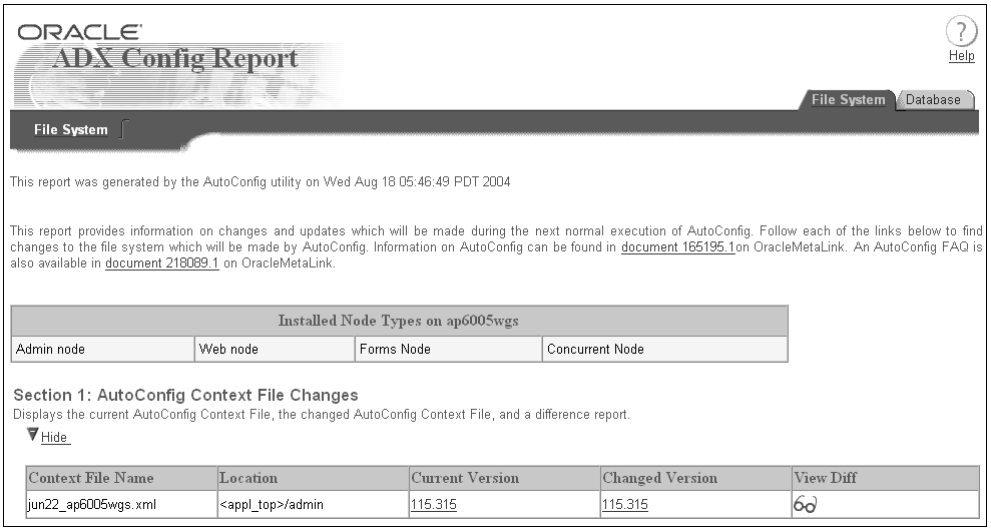
```
$ <RDBMS ORACLE_HOME>/appsutil/bin/adchkcfg.sh
```

Windows:

The script prompts for the location of the database context file and the APPS password.

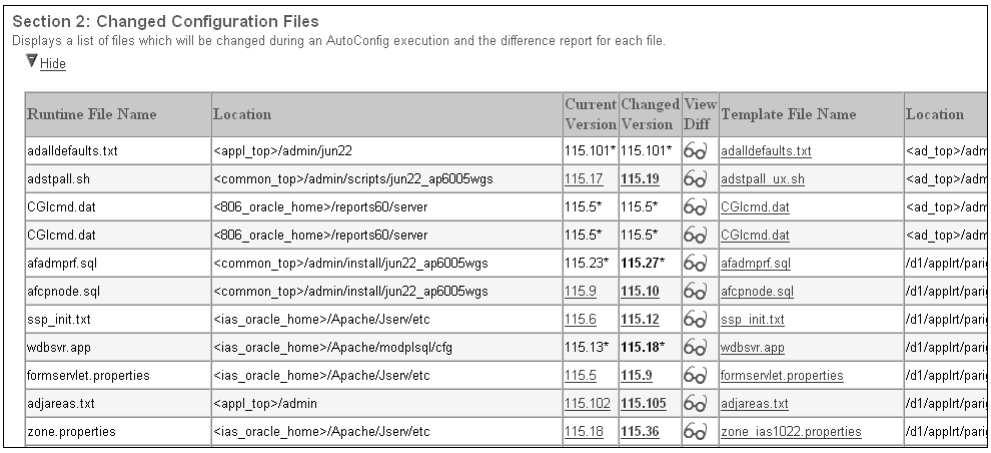
```
C:\> <RDBMS ORACLE_HOME>\appsutil\bin\adchkcfg.cmd
```

The AutoConfig test mode script produces a configuration report that shows the changes the AutoConfig script would have made. The configuration report, cfgcheck.html, is written to <APPL_TOP>/admin/<CONTEXT_NAME>/out/<MMDDhhmm> for the application tier, and for the database tier in <RDBMS_ORACLE_HOME>/appsutil/out/<CONTEXT_NAME>/<MMDDhhmm>. MMDDhhmm stands for the month, day, hour, and minute of the AutoConfig test mode script session.



The configuration report contains four sections. The first section lists the name, location and version number of the existing context file and the version number of the changed context file. Click the View Diff icon to see the differences between the existing and changed context files.

The second section lists the configuration files that would have been updated and contains a link to a difference report for each file.



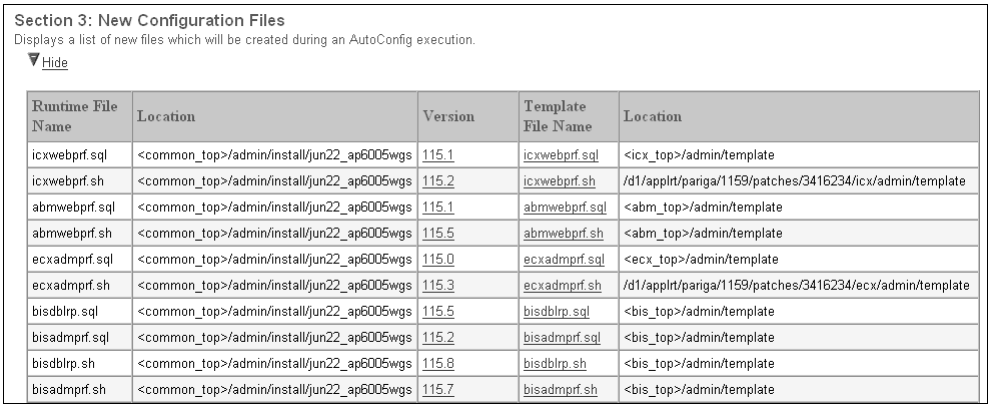
Section 2: Changed Configuration Files

Displays a list of files which will be changed during an AutoConfig execution and the difference report for each file.

Hide

Runtime File Name	Location	Current Version	Changed Version	View Diff	Template File Name	Location
adalldefaults.txt	<appl_top>/admin/jun22	115.101*	115.101*		adalldefaults.txt	<ad_top>/adm
adstpall.sh	<common_top>/admin/scripts/jun22_ap6005wgs	115.17	**115.19**		adstpall_ux.sh	<ad_top>/adm
CGIcmd.dat	<806_oracle_home>/reports60/server	115.5*	115.5*		CGIcmd.dat	<ad_top>/adm
CGIcmd.dat	<806_oracle_home>/reports60/server	115.5*	115.5*		CGIcmd.dat	<ad_top>/adm
afadmprf.sql	<common_top>/admin/install/jun22_ap6005wgs	115.23*	**115.27***		afadmprf.sql	/d1/applrt/pari
afcpnode.sql	<common_top>/admin/install/jun22_ap6005wgs	115.9	**115.10**		afcpnode.sql	/d1/applrt/pari
ssp_init.txt	<ias_oracle_home>/Apache/Jserv/etc	115.6	**115.12**		ssp_init.txt	/d1/applrt/pari
wdbsvr.app	<ias_oracle_home>/Apache/modplsql/cfg	115.13*	**115.18***		wdbsvr.app	/d1/applrt/pari
formservlet.properties	<ias_oracle_home>/Apache/Jserv/etc	115.5	**115.9**		formservlet.properties	/d1/applrt/pari
adjareas.txt	<appl_top>/admin	115.102	**115.105**		adjareas.txt	/d1/applrt/pari
zone.properties	<ias_oracle_home>/Apache/Jserv/etc	115.18	**115.36**		zone_ias1022.properties	/d1/applrt/pari

The third section lists the new configuration files that would have been created and contains a link to each new file.



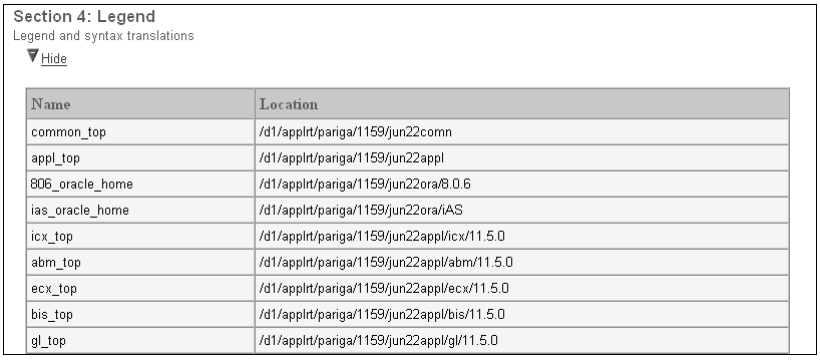
Section 3: New Configuration Files

Displays a list of new files which will be created during an AutoConfig execution.

Hide

Runtime File Name	Location	Version	Template File Name	Location
icxwebprf.sql	<common_top>/admin/install/jun22_ap6005wgs	115.1	icxwebprf.sql	<icx_top>/admin/template
icxwebprf.sh	<common_top>/admin/install/jun22_ap6005wgs	115.2	icxwebprf.sh	/d1/applrt/pariga/1159/patches/3416234/icx/admin/template
abmwebprf.sql	<common_top>/admin/install/jun22_ap6005wgs	115.1	abmwebprf.sql	<abm_top>/admin/template
abmwebprf.sh	<common_top>/admin/install/jun22_ap6005wgs	115.5	abmwebprf.sh	<abm_top>/admin/template
ecxadmprf.sql	<common_top>/admin/install/jun22_ap6005wgs	115.0	ecxadmprf.sql	<ecx_top>/admin/template
ecxadmprf.sh	<common_top>/admin/install/jun22_ap6005wgs	115.3	ecxadmprf.sh	/d1/applrt/pariga/1159/patches/3416234/ecx/admin/template
bisdblrp.sql	<common_top>/admin/install/jun22_ap6005wgs	115.5	bisdblrp.sql	<bis_top>/admin/template
bisadmprf.sql	<common_top>/admin/install/jun22_ap6005wgs	115.2	bisadmprf.sql	<bis_top>/admin/template
bisdblrp.sh	<common_top>/admin/install/jun22_ap6005wgs	115.8	bisdblrp.sh	<bis_top>/admin/template
bisadmprf.sh	<common_top>/admin/install/jun22_ap6005wgs	115.7	bisadmprf.sh	<bis_top>/admin/template

The fourth section displays important directory locations in your system.



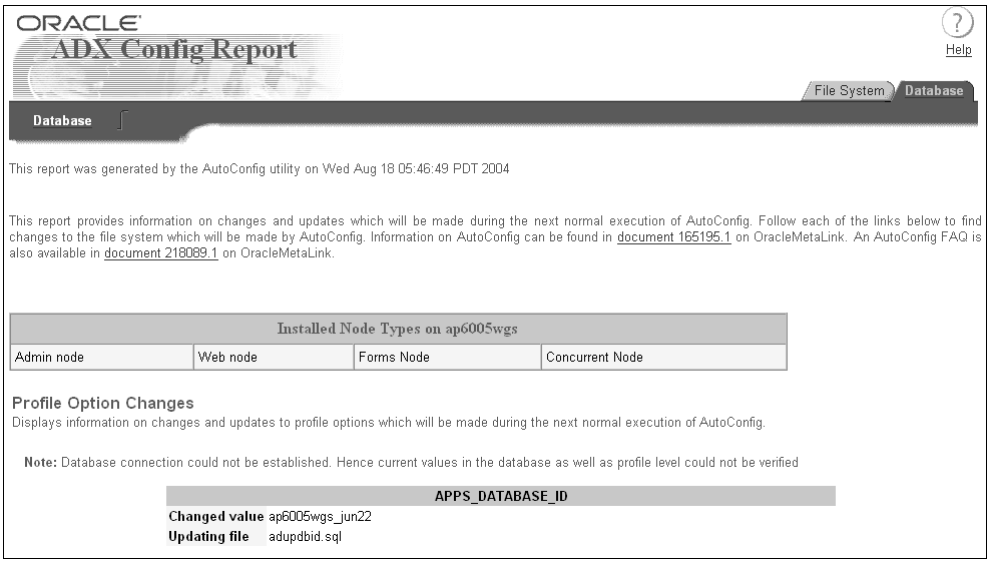
Section 4: Legend

Legend and syntax translations

Hide

Name	Location
common_top	/d1/applrt/pariga/1159/jun22comn
appl_top	/d1/applrt/pariga/1159/jun22appl
806_oracle_home	/d1/applrt/pariga/1159/jun22ora/8.0.6
ias_oracle_home	/d1/applrt/pariga/1159/jun22ora/iAS
icx_top	/d1/applrt/pariga/1159/jun22appl/icx/11.5.0
abm_top	/d1/applrt/pariga/1159/jun22appl/abm/11.5.0
ecx_top	/d1/applrt/pariga/1159/jun22appl/ecx/11.5.0
bis_top	/d1/applrt/pariga/1159/jun22appl/bis/11.5.0
gl_top	/d1/applrt/pariga/1159/jun22appl/gl/11.5.0

The configuration report for the database tier displays a list of profile options that would have been updated.



License Manager

License Manager registers the *licensing status* of products, country-specific functionalities, and languages that have been newly licensed (or downloaded) since the initial installation. You use it to register these components as being active in your system. This alerts the Applications utilities, such as AD Administration and AutoPatch, so that they will include these components, along with the existing components, in maintenance processes.

> **Note:** License Manager *does not* set up license agreements or determine pricing. The only way you can set up a license agreement is to purchase Applications product licenses through the Oracle Store or through an Oracle sales representative.

Base Language

As a part of the language registration, you can also use License Manager to indicate a change in the base language you use for business transactions.

Reports

License Manager can generate reports that list details like installed status and product abbreviation for the products and product-related components in your system. You can request detailed and summary reports for:

- Licensed Products/Shared Products
- Country-specific Functionalities
- Languages

The License Manager Interface

License Manager is a web-based utility in Oracle Applications Manager. It consists of these pages: the License Manager home page, and a group of licensing pages and reports pages. Since this utility is accessed through OAM, all the pages share a uniform look and feel.

Additional Information: See OAM Interface in Chapter 1.

Home page

Supplies the links to the following pages: License Products, License Country-specific Functionalities, and License Languages. Also provides links to the following reports: License Products, Shared Products, Country-specific Functionalities, Languages, and a Summary report page.

License Products

Provides options to register products by groups (E-Business Suite or Component Applications) or individually. Based on the selection, the appropriate pages appear.

License Country-specific Functionalities

Register additional products that have been localized to meet country-specific requirements.

License Languages

Register additional languages as active, and indicate a new base language.

Reports

Access reports about products, country-specific functionalities, and languages.

Additional Information: Click the OAM help button and navigate to the License Manager topic for more details about the pages.

Accessing License Manager

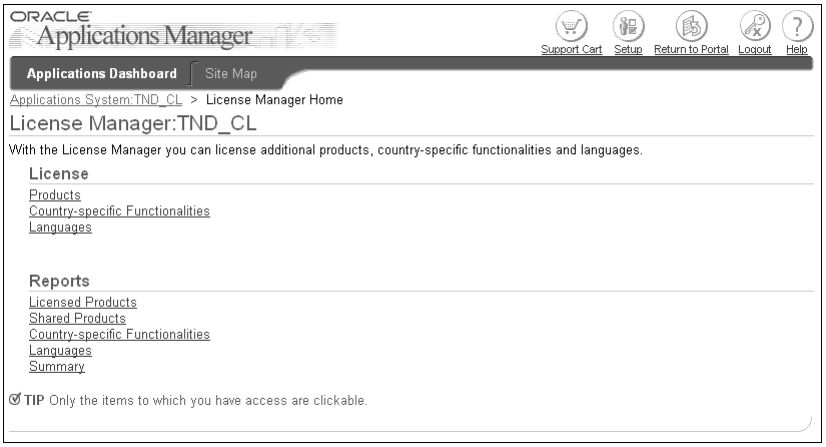
To access License Manager, log in to Oracle Applications Manager and choose License Manager from the Site Map.

1. Log in to Oracle Applications Manager.

   Follow the instructions in Accessing OAM in Chapter 1. From the Applications Dashboard, click the Site Map tab.

2. Go to the License Manager home page.

   From the Site Map, License Manager is on the Administration tab under the System Configuration heading. Click the License Manager link to go to the home page.



3. From this page, select one of the licensing (registration) options or report options by clicking the associated link.
4. Complete the information to register products, country-specific functionalities, or languages added since the initial installation of Oracle Applications. If you want to view a report, select a report type from the list and then complete the selection criteria.

> **Note:** Once a product or component has been registered, you cannot use License Manager to delete the registration.

5. Close License Manager.

   Once you have successfully completed the registration information, or you have finished viewing reports, click the Site Map tab to access other utilities, or click Logout to close OAM.

> **Additional Information:** See Registering Additional Applications Components in *Oracle Applications Maintenance Procedures* for detailed instructions for using License Manager.

AD Splicer

AD Splicer performs the same product registration function as License Manager. However, it registers *off-cycle* products — those that are released between maintenance packs — as active in your system. This process of "splicing" modifies the APPL_TOP and database so that AutoPatch and AD Administration recognize the off-cycle product as a valid product for a specific release.

> **Note:** AutoUpgrade ignores products for an existing release that have been added by AD Splicer. In addition, you cannot use AD Splicer to add custom products.

Patches that contain off-cycle products also contain the control files that AD Splicer needs to register the product. The patch also contains a readme file that describes how to install the new product(s).

Control Files

There are two kinds of AD Splicer control files: *product definition and product configuration.* You must customize the product configuration file, then copy it and the product definition file to APPL_TOP/admin before you run AD Splicer.

Product Definition Files

There are two product definition files per spliced product: <prod>prod.txt and <prod>terr.txt. These files define the product and the associated language information and must not be edited. For example, the product definition files for Oracle Sales Analyzer (zsa) are zsaprod.txt and zsaterr.txt.

Product Configuration

The newprods.txt file acts as a template to define necessary parameters for a spliced product. Here is an example of the product configuration file for Oracle Sales Analyzer (zsa):

```
product=zsa
base_product_top=*APPL_TOP*
oracle_schema=zsa
sizing_factor=100
main_tspace=*Product_Name*D
index_tspace=*Product_Name*X
temp_tspace=*Temporary_Tablespace*
default_tspace=*Product_Name*D
```

You may need to edit some of the values for the parameters in this file. Refer to the following table for more information. Do not change the order of the entries in the product configuration file — they must appear exactly as shown in the example.

Parameter	Description
product=	**Do not edit this entry**. The product abbreviation <prod> is already set, and must match the <prod>prod.txt and <prod>terr.txt control files for this product. Most internal references use <prod>.
base_product_top=	Identifies the base directory that contains the product's files. The default value, *APPL_TOP*, means the product's files are written in the directory your APPL_TOP environment is set to. If you want to write the product files to another directory, replace the *APPL_TOP* value with the full directory path.
oracle_schema=	Identifies the Oracle schema where database objects for the product are created. The default Oracle schema is the same as the product abbreviation. You can change this if you want to put the product's database objects in a different schema. **Moving a product's objects from one schema to another involves export/import and updates to internal Oracle Applications tables, so choose your initial schema carefully**.
sizing_factor=	Identifies the sizing factor Oracle Applications uses when creating tables and indexes for this product. The default value of 100 means 100%.The product's tables and indexes are created with the default sizes determined by Oracle. **We recommend you accept the default sizing factor**.

Tablespaces

During an upgrade to 11.5.10, you had the choice of continuing to use the existing tablespace model (referred to as Optimal Flexible Architecture or OFA) or of migrating to the new, consolidated tablespace model (referred to as Oracle Applications Tablespace Model or OATM). If you chose to continue to use OFA, you must create the appropriate tablespaces and indexes for the new product, and identify these tablespaces in newprods.txt.

> **Note:** If you chose the new tablespace model, AD Splicer does not need this information and ignores these parameters. For more information, see Oracle*MetaLink* Doc ID: 248857.1. See also Tablespace Management in *Oracle Applications Concepts*.

Parameters for identifying tablespaces are listed and described in the following table.

Parameter	Description
main_tspace=	Specifies the tablespace where this product's tables are created. Create a new tablespace called <prod>D (where <prod> is the product abbreviation listed in the product= line) to hold the tables for this new product. Then, replace *Product_Name* with the product abbreviation. For example, ZSAD.
index_tspace=	Specifies the tablespace where this product's indexes are created. Create a new tablespace called <prod>X (where <prod> is the product abbreviation listed in the product= line) to hold the indexes for this new product. Then, replace *Product_Name* with the product abbreviation. For example ZSAX.

Parameter	Description
temp_tspace=	Specifies the tablespace used by this product for creating temporary segments. In general, each Oracle Applications database should have a separate tablespace dedicated to temporary segments, and all Oracle Applications schemas (including the APPS schema) should use this tablespace. Replace *Temporary_Tablespace* with the value of the temporary tablespace for the Oracle Applications database in which you are installing this product. For example, TEMP.
default_tspace=	Specifies the default tablespace where this product's objects are created. The default tablespace is used if scripts create tables or indexes for this product without explicitly specifying a tablespace. Set this parameter to the same value you used for the main_tspace= line. For example, ZSAD.

The AD Splicer Interface

AD Splicer is a command line utility. It does not use menus or input screens.

Running AD Splicer

All the steps necessary to prepare for using this utility are described in Adding Off-cycle Products in Chapter 1 of *Oracle Applications Maintenance Procedures*. When instructed to do so, run AD Splicer as follows:

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Start AD Splicer

Start AD Splicer with this command:

UNIX:

```
$ cd $APPL_TOP/admin
$ adsplice
```

Windows:

```
C:\> cd %APPL_TOP%\admin
C:\> adsplice
```

You must run AD Splicer for each APPL_TOP and database combination so that the Applications utilities recognize the product as being spliced properly into the database.

File Character Set Converter

The File Character Set Converter converts individual files (one at a time) from one character set to another. You may need to perform this task to convert text files you receive from Oracle to the character set used by your system. Examples of files you might need to convert include SQL*Plus scripts, PL/SQL scripts, loader files, driver files, ODF files, header files, and HTML files.

Typically, you don't need to run this utility manually because AD Administration, AutoPatch, and Rapid Install do all required character set conversion for you automatically.

Required Parameters

The following parameters are required for running the converter.

Parameter	Definition
source_file	Path and file name for the (source) file to be converted.
source_char_set	Character set for the file to be converted (source).
destination_file	Path and file name for the (destination) file after it is converted.
dest_char_set	Character set for the converted (destination) file.

The File Character Set Converter Interface

The File Character Set Converter is a command line utility. It does not use menus or input screens.

Running the File Character Set Converter

To run this utility, complete these steps:

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Start the utility

Start the File Character Set Converter with this command:

```
adncnv <source_file> <source_char_set> <destination_file> <dest_char_set>
```

The path and file name for the source and the destination files can be the same if the source file's directory and the APPLTMP directory are on the same file system. In general, it is simpler and safer to use different source and destination file names.

> **Note:** If you cannot convert to the same file name, convert to a different file name or change APPLTMP to a directory on the same file system as the source file directory.

For example, to convert the file afdict.ldt from the we8iso8859p1 character set to the utf8 character set, you would type:

UNIX:

```
$ cd $FND_TOP/patch/115/import/<language>
$ cp afdict.ldt afdict.old
$ adncnv afdict.old we8iso8859p1 afdict.ldt utf8
```

Windows:

```
C:\> cd %FND_TOP%\patch\115\import\<language>
C:\> copy afdict.ldt afdict.old
C:\> adncnv afdict.old we8iso8859p1 afdict.ldt utf8
```

4

Maintenance

In order to ensure that your Oracle Applications system runs smoothly, you must perform routine maintenance tasks. This chapter discusses the AD utilities designed to help you perform those tasks. It contains these sections:

- About System Maintenance
- AD Administration
- AD Relink

About System Maintenance

After your system is installed, it will be necessary to perform certain maintenance tasks to keep it running smoothly. For example, you will generate form files, maintain snapshot information, relink executables, compile or validate the APPS schema, and so on. Some tasks are routine, and should be performed on a regular basis. Other tasks are performed more infrequently.

You run maintenance tasks from the command line using AD Administration. Once you start this utility, it presents the tasks in menu form, grouped generally by type of activity you will perform. For example, the tasks associated with compiling and reloading Applications database entities are grouped on the same menu.

In addition to the AD Administration maintenance tasks, this chapter describes AD Relink — a command line utility used to relink AD executables. While you can relink Oracle Applications product executables using AD Administration, you cannot relink AD utilities executables using AD ADministration, so you must relink them manually using the command line utility AD Relink.

AD Administration

AD Administration manages most of the maintenance tasks required for your Oracle Applications system. Currently, these maintenance tasks are grouped by the following types: generating Applications files, maintaining Applications files, compiling and reloading Applications database entities, maintaining Applications database entities, and enabling maintenance mode.

When you start AD Administration from the command line, it prompts you for the basic system-specific information it needs. For example, you need to supply a name for the log file where processing actions and error messages will be recorded.

Additional Information: See Prompts in Chapter 1.

Once you respond to these prompts, AD Administration displays the main menu, which serves as the gateway to various submenus where you select the individual maintenance tasks. For example, on the Generating Applications Files menu, you can run tasks that generate message files, form files, report files, graphics files, or product JAR files. These submenu tasks may also require you to respond to prompts to collect task-specific information. For example, some tasks require you to enter the number of workers you want to employ to process the jobs associated with the task.

> **Additional Information:** See Processing Tasks in Parallel in Chapter 1.

When you respond to AD Administration prompts, you are running the utility interactively. However, like AutoPatch and AD Controller, you can also run AD Administration non-interactively — specifying a previously created defaults file that contains the information necessary to run a specific maintenance task without user intervention.

> **Additional Information:** See Interactive and Non-interactive Processing in Chapter 1.

Prompts

In addition to the basic prompts described in Chapter 1, AD Administration may require additional information that is specific to one of the submenu tasks. If so, it displays additional prompts. For example, when running the Compile APPS schema task from the Compile/Reload Applications Database Entities submenu, AD Administration prompts you as follows:

```
Run Invoker's Rights processing in incremental mode [No]:
```

The task-specific prompts are described more fully in the discussion of each task.

Preparing for Non-interactive Processing

A discussion of command line prompts assumes you are running AD Administration interactively — you respond to the standard prompts and those required for specific tasks you choose from the AD main menu and submenus. AD Administration can also run some tasks non-interactively by using the information you store in a defaults file, instead of requiring you to respond to prompts.

> **Additional Information:** See Interactive and Non-interactive Processing on page 1-5. See also Scheduling Non-interactive Maintenance in *Oracle Applications Maintenance Procedures*.

Specifying a Menu Option in the AD Administration Defaults File

The same defaults file can be used to run different AD Administration tasks — a single file can contain all your choices for the different menu options. In order to choose which task the defaults file will run, you add *menu_option*= <menu choice> to the utility start command. This overrides any menu-specific key stroke information stored in the defaults file initially, and allows you to use the defaults file for any of the AD Administration menu items. It also ensures that the menu option you intended for the defaults file is always valid, even if the menu items are renumbered or relocated in subsequent releases.

The following table lists the menu options and the corresponding menu tasks:

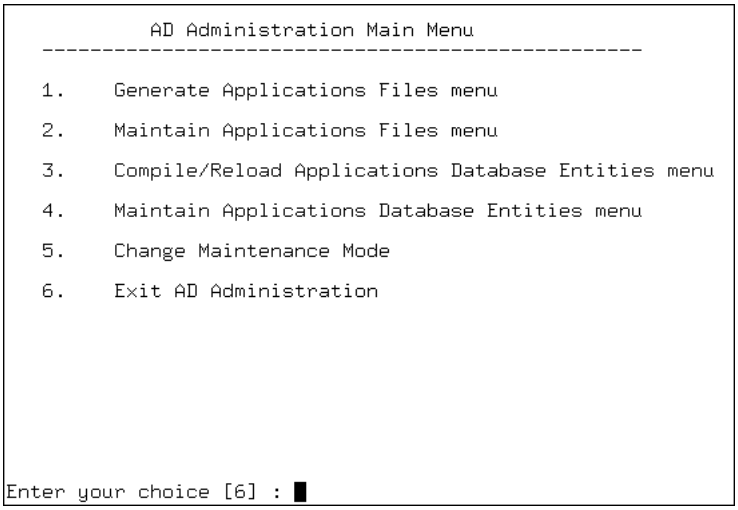
Menu Option	Corresponding AD Administration Menu Choice
GEN_MESSAGES	Generate message files
GEN_FORMS	Generate form files
GEN_GRAPHICS	Generate graphics files
GEN_REPORTS	Generate reports files
GEN_JARS	Generate product JAR files
RELINK	Relink Applications programs
CREATE_ENV	Create Applications environment file
COPY_FILES	Copy files to destinations
CONVERT_CHARSET	Convert character set
SCAN_APPLTOP	Scan the APPL_TOP for exceptions
SCAM_CUSTOM_DIR	Scan a CUSTOM directory for exceptions
LIST_SNAPSHOT	List snapshots
UPDATE_CURRENT_VIEW	Update current view snapshot
CREATE_SNAPSHOT	Create named snapshot
EXPORT_SNAPSHOT	Export snapshot to file
IMPORT_SNAPSHOT	Import snapshot from file
DELETE_SNAPSHOT	Delete named snapshot
CHECK_FILES	Check for missing files
CMP_INVALID	Compile APPS schema
CMP_MENU	Compile menu information
CMP_FLEXFIELDS	Compile flexfield data in AOL tables
RELOAD_JARS	Reload JAR files to database
VALIDATE_APPS	Validate APPS schema
CREATE_GRANTS	Recreate grants and synonyms for APPS schema
MAINTAIN_MLS	Maintain multi-lingual tables
CHECK_DUAL	Check DUAL table
MAINTAIN_MRC	Maintain Multiple Reporting Currencies schema
CONVERT_MCURR	Convert to Multiple Reporting Currencies
CONVERT_MULTI_ORG	Convert to Multi-Org
ENABLE_MAINT_MODE	Enable Maintenance Mode
DISABLE_MAINT_MODE	Disable Maintenance Mode

The AD Administration Interface

You start AD Administration from the command line. However, all maintenance tasks are initiated from the AD Administration Main Menu. This section describes some of the common features used to run this utility.

Main Menu

After you start AD Administration and respond to the prompts, the AD Administration Main Menu appears.



This menu displays the submenus where the individual maintenance tasks are grouped. To choose a submenu, type the number of the menu at the prompt. To exit AD Administration, press [Return].

Option Numbers

The submenus for AD Administration may display slightly different option names and numbers from the ones shown here. This is because when you start AD Administration, it "customizes" the menus based on the information it finds in the system configuration files. For example, if your system is already using Multi-Org functionality, the Maintain Applications Database Entities submenu does not display the Convert to Multi-Org task as an option.

Running AD Administration Interactively

Complete the steps in this section to display the AD Administration Main Menu and access the submenus and the maintenance tasks.

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Start AD Administration

From any directory, start AD Administration with this command:

```
$ adadmin
```

The utility starts and displays the first prompt.

Step 3 Respond to prompts

Complete the information in the AD Administration prompts. The basic AD command line prompts are described in Chapter 1. Prompts that are unique to options are described in this chapter in the section that describes the option.

When you complete the prompts, the Main Menu appears.

Step 4 Choose maintenance tasks

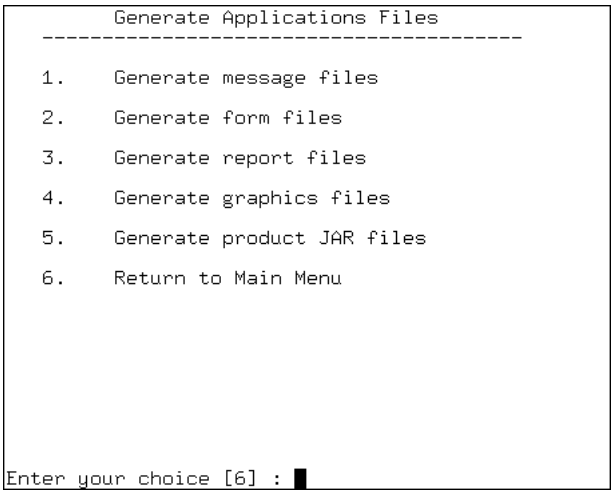
On the Main Menu, choose a submenu. The submenus and the options they display are described fully beginning with Generating Applications Files in the next section.

Step 5 Exit AD Administration

You can exit AD Administration from the Main Menu by choosing option 6 (Exit AD Administration) at the screen prompt. You can also choose to exit the utility at any prompt by typing *abort* on the command line. See Restart Files in Chapter 1 for information about restarting AD utilities after using the *abort* command.

Generating Applications Files

You may need to generate Applications files from time to time during your Applications life cycle. You access the associated tasks from the Generate Applications Files menu.



If system users are having difficulty accessing messages, forms, reports, or graphics, you may be able to resolve the issue by generating the associated files. Or, when you apply a patch that adds or changes product functionality, you may want to generate the associated files after you apply the patch, instead of running the generate driver during the patching downtime. The generate files tasks may be performed on any server, as required.

> **Note:** You do not have to shut down your system to generate files. However, users that access the files being generated (for example, for Human Resources forms) must log off.

> **Additional Information:** See Generating Product Files in *Oracle Applications Maintenance Procedures.*

Generate message files

Oracle Applications uses these files to display messages. This task generates message binary files (extension .msb) from Oracle Application Object Library tables.

Caution: Run this task only when instructed to do so in a patch readme file, or by Oracle Support Services.

Generate form files, report files, and graphics files

These tasks operate in much the same way.

- Generate form files

  Generates executable Oracle form files (extension .fmx) from the binary forms definition files (extension .fmb). The definition files are located under AU_TOP, and the executable files are stored under each product's directory.

- Generate report files

  Generates the binary Oracle Reports report files (extension .rdf).

- Generate graphics files

  Generates the Oracle Graphics files (extension .ogd) from the graphics definition files (extension .ogx).

The prompts and behavior work in similar fashion, except as noted:

- Ask for the number of workers and generate selected objects for select products in parallel.
- Display the current character set (from NLS_LANG) and ask if you want to generate forms, reports, or graphics objects in this character set.
- Ask if you want to regenerate Oracle Forms PL/SQL library files, menu files, and executable files. (**Forms files only**)
- Ask for the products associated with the forms, reports, or graphics objects.
- Ask if you want to generate specific forms, reports, or graphics objects for each selected product.
- Display the current set of installed languages and ask if you want to generate forms, reports, or graphics files in these languages.
- Create a list of all objects to generate.
- Display the list of objects to be generated. (**Specific objects or all objects**.)

Generate product JAR files

Generate Java archive (JAR) files whenever you upgrade the Developer6*i* technology stack or when recommended by Oracle Support Services. It signs JAR files (if on the Web server) and generates or recreates the following:

- Generates product JAR files in JAVA_TOP and copies them to APPL_TOP.
- Generates other Java-related files under APPL_TOP and JAVA_TOP.
- Recreates Java libraries (appsborg.zip and appsborg2.zip) under APPL_TOP and JAVA_TOP.

When you run the task, it prompts:

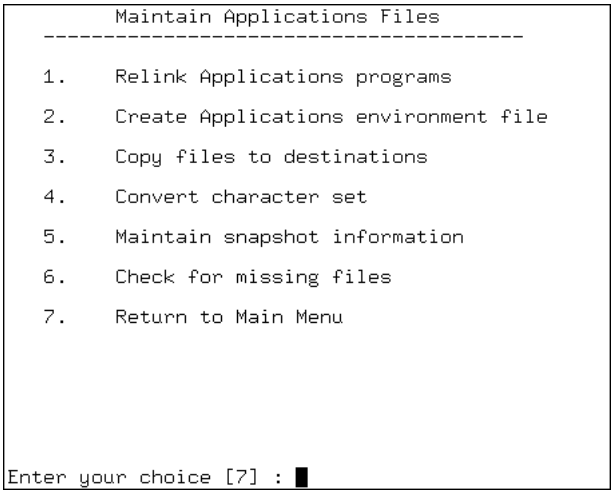
```
Do you wish to force generation of all jar files? [No]
```

If you choose No, it generates only JAR (Java archive) files that are missing or out-of-date. If you choose Yes, all JAR files are generated.

Note: If AD Administration displays a list of warnings or errors and objects that did not generate successfully and asks if you want to continue as if successful, review the log file to determine if the problems require attention. If you choose not to continue and restart your session at a later time, AD Administration attempts to regenerate only the files that did not generate successfully.

Maintaining Applications Files

Certain maintenance tasks are required to keep your Applications files up to date. For example, you may need to copy product files to a central location or convert files in the APPL_TOP to another character set. These tasks are grouped on the Maintain Applications Files menu.



Some of these tasks are performed for you without having to run AD Administration. For example, each time you run AutoConfig, it creates the environment file. However, when required, you can run any of these tasks by choosing it from this menu. They may be performed on any server, as required.

Relink Applications programs

Relinks Oracle Applications executable programs with the Oracle server libraries so that they function with the Oracle database. For each product, choose whether to link all executables or only specific ones.

Note: The default relinks without debug information. Use the debug option only when requested to do so by Oracle Support Services.

Additional Information: AD Administration cannot link executables for AD products. See Relinking AD Executables in *Oracle Applications Maintenance Procedures.*

Create Applications environment file [subkey in registry]

Creates an environment file (UNIX) or an environment subkey (Windows registry) that defines your system configuration.

> **Note:** If your system was installed at release 11.5.8 or later, AutoConfig is enabled. AD Administration prompts for the environment file name (<CONTEXT_NAME>.env) and generates the environment file. If you have not enabled AutoConfig, see Oracle*MetaLink* Doc ID: 165195.1.

The environment file is placed directly under the APPL_TOP (UNIX) or under the \\HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\APPLICATIONS\11.5.10 subkey (Windows registry).

If you need to customize this file, you can do so using AutoConfig in the Oracle Applications Manager.

> **Additional Information:** See Managing Configuration Parameters in *Oracle Applications Maintenance Procedures*.

Copy files to destinations

Copies files from each product area to central locations where they can be easily referenced by non-Applications programs. This option uses revision-based copy logic to ensure that the destination file versions are the same as, or higher than, the source file versions.

> **Note:** We recommend that you do not use the "force" option to overwrite existing files unless instructed by Oracle Support Services. Copying files with this option updates all JAR files. JInitiator then downloads required JAR files to each client again, causing runtime performance degradation.

The file types and their respective destinations are shown in the following table:

These files:	...are copied to (UNIX)	...are copied to (Windows)
Java files	$JAVA_TOP	%JAVA_TOP%
HTML files	$OAH_TOP	%OAH_TOP%
Media files	$OAM_TOP	%OAM_TOP%

The directories for the variables are specified in the adovars.env file (UNIX) or the adovars.cmd file (Windows).

> **Note:** When this option is used to copy reports or graphics files, the default destination is under AU_TOP.

Convert character set

Prepares the files in the APPL_TOP for conversion to another character set, and then performs the conversion.

> **Additional Information:** See Internationalization Support in *Oracle Applications Concepts*. See also Oracle*MetaLink* Doc ID: 124721.1.

When you choose this option, AD Administration presents another submenu, which contains options for scanning your files in preparation for the conversion. The scan searches for exceptions — files that will have incomplete (*lossy*) conversions — so that you can fix potential problems before you actually convert the character set. Choose one of the following scan options.

> **Note:** Verify the compatibility of the database character set before converting the APPL_TOP character set.

1. Scan the APPL_TOP for exceptions.

   Scans the APPL_TOP and creates three files in the admin\<SID>\out directory.

File	Contents
admanifest_excp.lst	Lists files that will not be converted because of lossy conversion.
admanifest.lst	Lists files that can be converted.
admanifest_lossy.lst	Lists files with lossy conversions, including line by line detail.

   Review the files listed in admanifest_excp.lst. Fix files that report lossy conversion before you convert the character set. Repeat this task until there are no entries in admanifest_excp.lst. If you need to see more detail, review admanifest_lossy.lst.

2. Scan a CUSTOM directory for exceptions.

   Collects the same information as the first task, but scans custom Applications directories rather than the APPL_TOP directory.

3. Convert character set.

   Run this task only if admanifest_excp.lst has no entries. It prompts you for the manifest file (admanifest.lst) created when you ran the scan option(s).

   The utility backs up the product source files and the APPL_TOP/admin source files. It saves product files in the <PROD>_TOP directories in the format <prod>_s_<char_set>.zip. It saves admin source files in the APPL_TOP/admin directory in the format admin_s_<char_set>.zip

Maintain snapshot information

There are two types of snapshots: APPL_TOP snapshots and global snapshots. An APPL_TOP snapshot lists patches and versions of files in the APPL_TOP. A global snapshot lists patches and latest versions of files in the entire Applications system (that is, across all APPL_TOPs).

Both APPL_TOP snapshots and global snapshots may be either *current view* snapshots or *named view* snapshots. A current view snapshot is created once and updated when appropriate to maintain a consistent view. A named view snapshot is a copy of the current view snapshot at a particular time (not necessarily the latest current view snapshot) and is not updated.

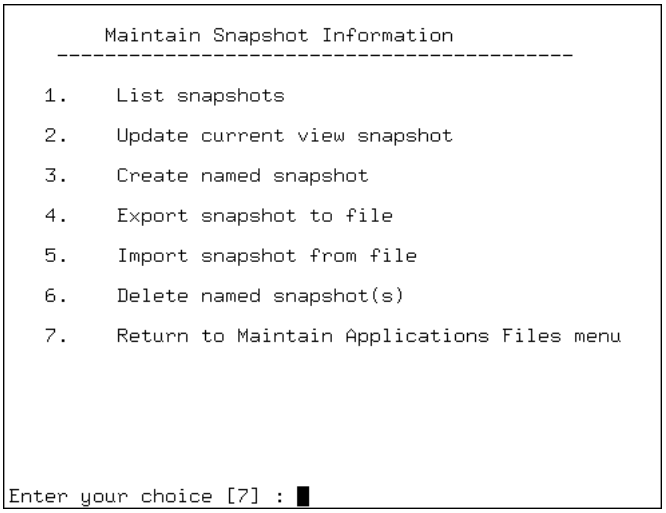
Patch Wizard uses the information contained in the global current view snapshot to determine which patches have already been applied. AutoPatch uses the APPL_TOP current view snapshot to determine if all prerequisite patches have been applied to that APPL_TOP. Snapshot information is stored in the AD_SNAPSHOTS, AD_SNAPSHOT_FILES, and AD_SNAPSHOT_BUGFIXES tables.

A complete current view snapshot is required for automatic prerequisite patch checking to operate. During the installation, Rapid Install created a current snapshot as a baseline. And, each time you run AutoPatch, it automatically creates a new (updated) snapshot so that the information is current as of the application of the patch.

However, after an upgrade to Release 11*i*, you must update the current snapshot by running this AD Administration task.

1. Access the Maintain Snapshot Information menu.

   Choose Maintain Snapshot Information from the AD Administration Main menu.



2. Choose an option.

   From this menu, you can:

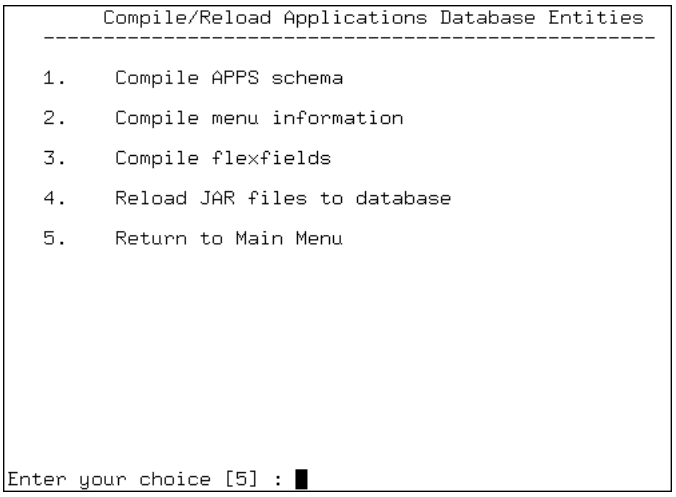
   - List snapshots (stored in the system)
   - Update current view snapshot (APPL_TOP and global)
   - Create named snapshot (select a current view snapshot to copy and name)
   - Export snapshot to file (select one to export to a text file)
   - Import snapshot from (a text) file
   - Delete named snapshot (select a snapshot for deletion)

Check for missing files

Verifies that all files needed to run Oracle Applications for the current configuration are in the current APPL_TOP. Choose this task if you suspect there are files missing in your APPL_TOP.

Compiling or Reloading Database Entities

Database entities are database objects or data in the database related to Oracle Applications. The tasks for compiling and reloading database entities are grouped on the Compile/Reload Applications Database Entities menu.



You run the tasks on this menu any time you need to compile or reload database objects. For example, after you upload new menu entries or apply a patch that changes the setup of flexfields. Run these tasks *only* on the node where the core AD technology directories (the administration server) are located.

Compile APPS schema

Spawns parallel workers to compile invalid database objects in the APPS schema. When you choose this task, AD Administration prompts:

```
Run Invoker's Rights processing in incremental mode [No] ?
```

Typing Yes at this prompt causes Invoker Rights processing to run only on packages that have changed since Invoker Rights processing was last run.

> **Note:** Invoker Rights processing does not modify any of your custom packages, procedures, or functions unless you imbed RCS header information (in the identical format to those found in Oracle Applications PL/SQL files) in the first five lines of your PL/SQL source text.

> **Additional Information:** See Compiling Invalid Objects in *Oracle Applications Maintenance Procedures*. See also Invoker Rights in *Oracle Applications Concepts*.

Compile menu information

Compiles menu data structures. Choose this task after you have uploaded menu entries to the FND_MENU_ENTRIES table, or if Compile Security concurrent requests submitted from the Menus form (after changing menu entries) fail for any reason.

AD Administration asks if you want to force compilation of all menus. If you choose the default (No), only menus with changes are compiled. If you enter Yes, all menus are compiled. Compiling all menus is generally not advised.

Compile flexfield data

Compiles flexfield data structures in Oracle Application Object Library (FND) tables. Choose this task after you apply a patch that changes the setup of flexfields. Patches usually indicate when you should perform this step.

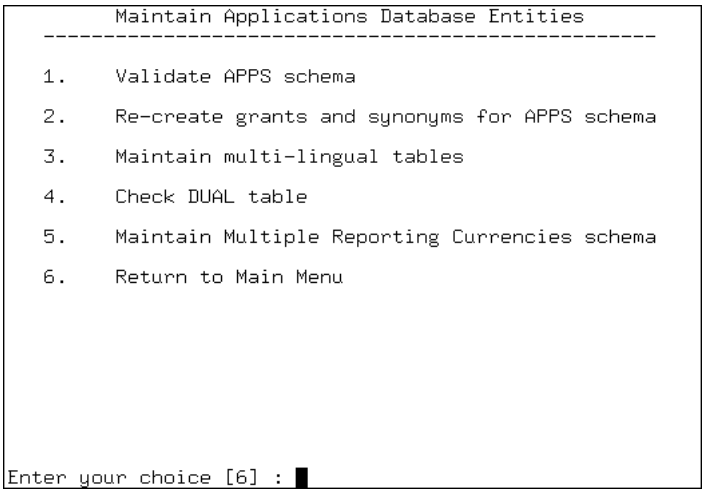
Flexfields automatically compile data when you use them for the first time, so running this task is not required. However, compiling flexfield data at a specific time (for example, when system use is low), rather than automatically at first use, can alleviate potential runtime performance issues.

Reload JAR files to database

Reloads all appropriate Oracle Applications JAR files into the database. Choose this task if all Oracle Applications Java classes are removed from your database, for example, if the database Java Virtual Machine (JVM) is reloaded because of a corrupt database.

Maintaining Applications Database Entities

During normal system use, the integrity of your database can be compromised, for example through user error or after you apply a large patch. It's a good idea to verify the integrity of database entities as a regular maintenance procedure, or whenever the behavior of your system indicates that database entities may have been corrupted. These maintenance tasks are grouped on the Maintain Applications Database Entities menu.



Some tasks on this menu report on issues, or potential issues, with database entities, and others actually remedy the issues. Run these tasks *only* on the node where the core AD technology directories (the administration server) are located.

Validate APPS schema

Verifies the integrity of the APPS schema. It produces a report named <APPS schema name>.lst that lists issues and potential issues, grouped by the action required:

- Issues you MUST fix (not specific to the APPS schema)
- Issues you MUST fix (specific to the APPS schema)
- Issues you may want to address (specific to the APPS schema)

The report is located in $APPL_TOP/admin/<SID>/out (UNIX), where <SID> is the value of the ORACLE_SID or TWO_TASK variable, or in %APPL_TOP%\admin\<SID>\out (Windows), where <SID> is the value of the LOCAL variable. Most issues can be fixed by either compiling invalid database objects or recreating grants and synonyms.

Additional Information: See Creating Grants and Synonyms and Compiling Invalid Objects in *Oracle Applications Maintenance Procedures*.

Note: It is no longer necessary to fix issues found in the first two sections of the report before you run the Maintain Multiple Reporting Currencies task.

Recreate grants and synonyms for APPS schema

This task recreates grants and synonyms for the Oracle Applications public schema (APPLSYSPUB), recreates grants on some packages from SYSTEM to APPS, and spawns parallel workers to recreate grants and synonyms linking sequences and tables in the base schemas to the APPS schema.

Typically, you run this task after the Validate APPS schema task has reported issues with missing grants and synonyms.

Note: It is no longer mecessary to run the Maintain MRC task to set up grants and synonyms after you fix issues with the APPS schema.

Maintain multi-lingual tables

Run this task after you add a language. It prompts you for the number of workers, then updates all multilingual tables.

Check DUAL table

Some Oracle Applications products must access the DUAL table. It must exist in the SYS schema and contain *exactly* one row. This tasks verifies the existence of this table and the single row.

Caution: If the DUAL table does not exist, or if it does not contain only one row, the Applications products that access it will fail.

Maintain [or Convert to] Multiple Reporting Currencies

If you have installed Multiple Reporting Currencies (MRC) functionality, this menu option is called Maintain Multiple Reporting Currencies schema. If you have not, it is called Convert to Multiple Reporting Currencies, which you use to install MRC.

Note: It is no longer necessary to validate the APPS schema before running this task. See *Multiple Reporting Currencies in Oracle Applications* for more information about MRC in this release.

MRC is implemented using the standard APPS schema. That means that the Maintain (or Convert to) MRC task only recreates MRC triggers in the APPS schema. It requires none of the other logic found in previous releases. Once AD Administration completes the MRC processing, check the log file (adadmin.log) for errors and warnings. Rerun this task until the report contains no issues.

Note: Shut down the concurrent managers before updating MRC.

Additional Information: See Converting to MRC in *Oracle Applications Maintenance Procedures.*

Convert to Multi-Org

Appears as a menu choice only if Multi-Org is *not* installed in your database. Use it to convert a standard product group into a Multi-Org product group with one operating unit defined at the site level. The Convert to Multi-Org task does the following:

- Asks for the number of parallel workers (and dynamically uses multiple workers on any Multi-Org partitioned table that has more than one million rows).
- Disables all enabled triggers in the APPS schema.
- Converts seed data and transaction data to Multi-Org in parallel.
- Re-enables all previously disabled triggers in the APPS schema.

Additional Information: See *Multiple Organizations in Oracle Applications*. See also Converting to Multi-Org in *Oracle Applications Maintenance Procedures.*

Changing Maintenance Mode

Maintenance mode affects the process of maintaining Oracle Applications and must be enabled to optimize performance and reduce downtime of AutoPatch sessions. You toggle maintenance mode from enabled to disabled by using the Change Maintenance Mode menu of AD Administration.

Additional Information: See Maintenance Mode in *Oracle Applications Concepts.*

When you choose Change Maintenance Mode from the Main Menu, the Change Maintenance Mode menu appears with the maintenance mode status displayed at the top.

```
            Change Maintenance Mode
   ----------------------------------------

Maintenance Mode is currently: Disabled.


Maintenance mode should normally be enabled when patching
Oracle Applications and disabled when users are logged on
to the system.  See the Oracle Applications Maintenance
Utilities manual for more information about maintenance mode.


Please select an option:

   1.    Enable Maintenance Mode

   2.    Disable Maintenance Mode

   3.    Return to Main Menu


Enter your choice [3] : 
```

Select option 1 to enable maintenance mode or option 2 to disable it. Maintenance mode must be enabled before running AutoPatch and disabled during normal runtime operation.

AD Relink

You use AD Relink to relink AD excutables with the Oracle server product libraries. Linking executables keeps them functioning properly with the Oracle database. While you link product executables using the Relink Applications Executables task on the AD Administration Maintain Applications Files submenu, you cannot use it to relink an AD executable. So, you must relink AD executables manually using AD Relink. You can relink multiple AD executables simultaneously.

Log Files

As you run AD Relink, it creates a log file (adrelink.log) where it records errors and messages. AD Relink appends information about the latest relink action to the end of the file. This file is located in APPL_TOP/admin/log. If an error occurs while you are using AD Relink, or if you are not sure that the relinking was successful, review this file to see what issues should be fixed.

> **Note:** Relinking errors encountered during an AD Administration or an AutoPatch session are recorded in the main log files for those utilities. See Log Files in Chapter 1.

To recover disk space, you can delete the adrelink.log file if you do not need the information. A new log file is created each time AD Relink runs.

Command Line Arguments

You can modify or refine the operation of AD Relink with the command line arguments in the following table. They are intended for use specifically for this utility.

force	Description
Purpose	indicates which executable programs to relink. **This argument is required**.
Values	n, meaning relink only if the libraries or object files are more recent that the current executable program. y, meaning relink regardless of the status of the libraries or object files.
Default	none (you must enter either y or n)
Example	adrelink force=n

backup_mode	Description
Purpose	indicates whether you want to back up executables
Values	none, meaning do not back up any executables all, meaning back up all executables file, meaning back up files according to instructions in adlinkbk.txt

backup_mode	Description
Default	backup_mode=file
Example	adrelink force=n backup_mode=all

Files that are critical to running Oracle Applications are listed in the adlinkbk.txt file, which is located in APPL_TOP/admin. Using the backup_mode=file argument directs AD Relink to back up only these files.

The AD Relink Interface

You run AD Relink from the command line. It does not use menus or input screens.

Running AD Relink

Run AD Relink as follows.

1. Set the environment.

   You must set the environment to indicate the location of the configuration parameters that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1.

2. Relink files.

   Run AD Relink with the following command:

   UNIX:

   ```
   S adrelink.sh force=n "ad <executable name>"
   ```

   Windows:

   ```
   C:\> sh adrelink.sh force=n "ad <executable name>"
   ```

   Additional Information: See Relinking AD Executables in *Oracle Applications Maintenance Procedures* for complete instructions for running this utility with the various command line arguments.

5

Patching

Applying a patch updates your existing system in various ways, from adding a new feature or product to improving system performance. This chapter describes the various types of patches, and the utilities you use to apply them and monitor their effect on your system. It contains these sections:

- About Patches and Patching
- Patch Types and Structure
- AutoPatch
- AD Merge Patch
- Patch Wizard

About Patches and Patching

Throughout the course of the Oracle Applications life cycle, patches are applied for a number of reasons, including:

- Updating to a higher maintenance level
- Applying the latest product enhancements
- Adding a new feature or functionality
- Fixing an existing issue

Based on the type of patch, it may update the file system, or the database, or both. Patches are applied and tracked as needed by using one of the utilities designed specifically for that purpose. Some of these utilities are run from the command line, and others are web-based.

Command Line Patching Utilities

The following utilities are run from the command line.

AutoPatch

Patches are necessary to update and enhance your Oracle Applications system. AutoPatch is the utility used to apply all patches to the Oracle Applications file system or database.

AD Merge Patch

When you apply patches individually, you must perform patching tasks multiple times. For example, for every individual patch there may be duplicate link, generate,

and connect processes. AD Merge Patch merges multiple patches into a single patch so that the required patching tasks and processes are performed only once.

Web-based Patching Utilities

The following utilities are web-based. You access them through the Oracle Applications Manager (OAM).

Patch Wizard

An important part of the patching process is to keep abreast of new patches that are recommended, and analyze their effects before you actually apply them. With Patch Wizard, you can determine patches that have not been applied to your system, but are recommended to keep the system current. It can also advise you about the effects on your system of applying an individual patch before you apply it.

Patch Types and Structure

Patches are defined by type and by format. The patch *type* describes the purpose of the patch. For example, a patch may add product functionality, or it may fix an existing issue. The patch *format* describes the way the patch is packaged and applied. For example, a stand-alone patch focuses on a single, specific issue, while a minipack is in essence a merged patch that consolidates all patches for a specific product for a specific period of time.

Note: All Applications patches are available from Oracle*MetaLink*.

This section also discusses the *structure* of an Oracle Applications patch, including the directory structure, and the driver files that perform the actions specified in the patch.

Patch Types

There are several types of patches that you may be asked to apply to your Oracle Applications system. They are described in the following table.

Patch Type	Description
Bug fix	Fixes an existing issue.
New feature	Adds new functionality.
Interoperability	Contains Oracle Applications files and database objects to make the current version of Oracle Applications compatible with a newer version of the database or a technology stack component. For example, to enable an Oracle 10g database to work with Oracle Applications Release 11.5.10.
Diagnostic	Released specifically to determine the source of an issue. A diagnostic patch does not contain fixes.
Translation	Contains Oracle Applications files that have been translated from English to another language. A translation patch may also execute tasks to load or update data in the database.
Upgrade	Fixes a problem with, or improves the performance of, an upgrade from a previous major release, such as 10.7 or 11.0.

Patch Type	Description
Documentation	Updates Oracle Applications Online Help. When applying a product minipack or a stand-alone patch that adds a new feature, review the Features Summary Matrices on Oracle*MetaLink* for the associated documentation patch.

Patch Formats

Patches are released in the following formats. If a patch format is described as *cumulative*, that patch contains a consolidation of updates from the inception of Release 11*i*, up to, and including, the latest release level.

Patches	Description
Stand-alone	A patch that fixes a specific issue or provides new functionality.
High-priority	A patch identified by Oracle Development as having an impact that is broad enough to merit application by all customers who have installed the affected product.
Minipack	A consolidation of all patches for a product. The naming convention is 11i.<product>.<minipack letter> such as 11i.AD.I. Minipacks with a higher <minipack letter>) supersede previous versions. They are cumulative.
Family pack	A consolidation of a set of minipacks and other patches for a product family. Family packs with a higher number supersede previous versions. They are cumulative.
Consolidated Update	An update containing generally recommended patches and additional targeted patches combined into a single patch. Applying a consolidated update brings a release to the latest recommended patch level.
Family consolidated upgrade patch	All *upgrade-related*, high-priority patches consolidated from all the products within a product family. Family consolidated upgrade patches are released as needed. The *Oracle Applications Release Notes* lists the most recent family consolidated upgrade patches.
Maintenance pack	A consolidation of all minipacks for all products. A maintenance pack updates a system to a new point release of Oracle Applications, such as Release 11.5.10. Maintenance packs with a higher number supersede previous versions. They are cumulative.

Patch File Structure

Patches generally consist of a top-level directory, several files in the top-level directory, and one or more subdirectories. The top-level directory is named <patchnum>, where <patchnum> is the number of the patch. The most important files in the top-level directory are: README.txt, README.html and the driver files (c<patchnum>.drv, d<patchnum>.drv, g<patchnum>.drv, and u<patchnum>.drv). For most patches, applying the patch drivers is the only action required.

The README.txt or README.html files for each patch describes what the patch does, lists files in the patch and indicates which servers to run the patch on. It also explains any steps required to completely apply the patch, including prerequisite patches or manual steps.

Patch Driver Files

A patch may contain one or more split patch driver files. A *split* driver means that patch actions are divided by function: a *copy driver* to change product files, a *database*

driver to change database objects, and a *generate driver* to generate new forms, messages, graphics, and other new objects. Unless you are instructed to apply only one driver, you always apply split patch drivers in the following order: copy driver, database driver, and generate driver.

In some cases, rather than split drivers, a patch may contain one *unified* driver. This driver contains the commands necessary to change files and database objects, and to generate new objects, in a single driver file.

> **Caution:** Patches must always be applied in their entirety. If you apply a patch to update the file system, you must also apply the corresponding database and generate portions (if any).

Copy Driver

The copy driver is named c<patchnum>.drv. When you apply the copy driver, AutoPatch performs the following actions:

- Extracts the appropriate files from each product's C library.
- Compares the extracted object modules with their corresponding files in the patch directory. It also makes this type of comparison with files such as forms, reports, and SQL scripts.
- Backs up any product file with a more recent version in the patch directory to a subdirectory in the patch directory. For example, if <patch_dir> is the patch directory, <system_name> is the Applications System name, <appl_top_name> is the APPL_TOP name, and <prod> is the name of the product being patched, it backs up:

  ```
  <PROD>_TOP/<subdir(s)>/<old_file_name>

  to

  <patch_dir>/backup/<system_name>/<appl_top_name>/ \
  <prod>/<subdir(s)>/<old_file_name>
  ```

  > **Note:** The Applications system name and the APPL_TOP name are determined during the Rapid Install process.

- Replaces each product's outdated files with newer files from the patch directory.
- Loads the new object modules into the C libraries.
- Relinks the Oracle Applications products with the operating system, Oracle server, and other Oracle products libraries.
- Applies changed Java class files and regenerates JAR files as needed.
- Copies any specified HTML or media files to their respective destinations.
- Compiles out-of-date Java Server Page (JSP) files (if any JSP files are included in the patch).

In a multi-node system, you run the copy driver on all APPL_TOP directories containing one or more of the files being replaced by the patch. If in doubt, run it on all APPL_TOP directories on all nodes. AutoPatch will determine the appropriate actions.

Additional Information: Applying Patches to a Multi-Node System in *Oracle Applications Maintenance Procedures.*

Database Driver

The database driver is named d<patchnum>.drv. Apply this driver *only* on the APPL_TOP of the node where the AD technology files (known as the administration server) are located. When you apply the database driver, AutoPatch performs these actions:

- Gets a list of current invalid objects in the APPS schema.
- Determines whether the action was performed in a previous patch.
- Runs SQL scripts and EXEC commands, which change Oracle Applications database objects. By default, AutoPatch runs scripts and commands in parallel.
- Performs Invoker Rights processing if the patch contains a package command.

Note: Invoker Rights processing will not modify any of your custom packages, procedures, or functions unless you imbed RCS header information (in the identical format to those found in Oracle Applications PL/SQL files) in the first five lines of your PL/SQL source text. See Oracle server documentation for more information.

- Compiles invalid objects in the database.
- Assembles a list of current invalid objects in the APPS schema.

Generate Driver

The generate driver is named g<patchnum>.drv. Apply this driver file on all APPL_TOP directories containing one or more files being generated by the patch. If in doubt, apply it to all APPL_TOPs on all nodes.

Unified Driver

The unified driver is named u<patchnum>.drv. This driver replaces the c<patchnum>.drv, d<patchnum>.drv, and g<patchnum>.drv drivers with a single driver file. AutoPatch runs only the actions that are required for the current APPL_TOP.

Note: Unified drivers will eventually be used by most Oracle Applications patches. AutoPatch and AD Merge Patch support patches with split drivers, and those with unified drivers.

AutoPatch

You use AutoPatch to apply patches to the Oracle Applications file system or database. It gathers necessary information about your system through a series of prompts Once you have completed the prompts, AutoPatch performs all the tasks required to apply the patch, including the following:

- Unloads patch metadata.
- Uploads patch information from a prior patch session to the database (if needed).
- Reads and validates the patch driver file and reads the product driver files.

- Compares version numbers of object modules from the product libraries and version numbers of the existing files against the patch files.
- Backs up all existing files that will be changed by the patch.
- Copies files.
- Archives files in libraries.
- Relinks executables.
- Generates forms, reports, message, graphics, and Java archive (JAR) files.
- Compiles JSP files and invalid database objects.
- Updates database objects.
- Maintains Multiple Reporting Currencies (MRC) schema, if needed.
- Runs AutoConfig to update configuration files, if any template files are introduced or updated by the patch.
- Saves patch information to the database.

AutoPatch takes no action if a patch contains no new updates to files or database objects in your system.

Preparing your System for Patching

Before you begin a patching session, there are some important tasks you need to complete.

Enable maintenance mode

Before you initiate an AutoPatch session, you need to shut down the Workflow Business Events System and set up function security so that no Oracle Applications functions are available to users. This ensures optimal performance and reduces downtime when applying a patch. Maintenance mode (new in 11.5.10), provides a clear separation between normal runtime operation of Oracle Applications and system downtime for maintenance.

During a maintenance mode downtime, user login is restricted. Users are redirected to a system downtime URL, which informs them that the maintenance session is in progress. The Oracle Applications Manager (OAM) Maintenance Mode page allows you to schedule system downtime and send alert messages to notify users of the downtime schedule.

To enable or disable maintenance mode, use the Change Maintenance Mode menu in AD Administration.

Caution: You can run AutoPatch by using *options=hotpatch* on the command line when maintenance mode is disabled. However, applying a "hot patch" causes significant performance degradation.

Additional Information: See Changing Maintenance Mode in Chapter 4 of this book.

Shut down services

If you are applying a patch that updates or relinks files, shut down the concurrent manager, Web server listeners, forms server listeners if the files are on a node that

contains the associated servers. For example, if the files are on the node that contains the concurrent processing server, shut down the concurrent managers.

> **Additional Information:** See Applying a Patch Interactively in *Oracle Applications Maintenance Procedures*.

Log Files

In addition to the main log file (adpatch.log), AutoPatch also creates several other log files for specific purposes. For example, some log files record all the actions associated with parallel workers. The log files are written to $APPL_TOP/admin/<SID>/log (UNIX), where <SID> is the value of your ORACLE_SID or TWO_TASK variable, or in %APPL_TOP%\admin \<SID>\log (Windows), where <SID> is the value of ORACLE_SID or LOCAL (Windows). Review the log files when the AutoPatch session is complete.

The log directory contains adpatch.log and adpatch.lgi, and may contain one or more additional files as described in the following table. If AutoPatch does not perform an action, it does not generate the log file associated with that type of action.

Log File	Description
adpatch.log	main AutoPatch log file (default name)
adpatch.lgi	for AutoPatch informational messages (default name)
adrelink.log	for relinking
adlibin.log	for moving C object files into a product's C library
adlibout.log	for moving C object files out of a product's C library
adwork*xxx*.log	for database operations run in parallel

Prompts

In addition to the standard prompts common to most AD utilities, AutoPatch also asks for information specific to the patching process. You must respond to all prompts for each driver you run. That means if a patch requires you to run a copy driver, a database driver, and a generate driver, you would run AutoPatch three times and respond to all prompts.

> **Caution:** Do not run multiple sessions of AutoPatch on the same Applications system at the same time.

Main log file name

The main AutoPatch log file is named adpatch.log by default. We recommend you change the name to indicate the associated driver file, using a .log extension. For example, if the database driver is d123456.drv, the log file should be d123456.log.

SYSTEM and AOL user passwords

AutoPatch prompts for the SYSTEM and AOL user passwords. By default, it does not validate the password information.

> **Note:** You can change this behavior by using *options=validate* on the command line. See Command Line Arguments on page 5-12.

Patch directory

AutoPatch asks you to specify the directory where the patch files have been unzipped. The default is the directory from which you started AutoPatch. If necessary, specify the full path name to the directory where you unzipped the patch files. The operating system user running AutoPatch must have write permissions to that directory.

Patch driver file

AutoPatch prompts for the name of the patch driver file. By default, it does not check the integrity of the patch — whether the version of each file referenced in a driver file copy action matches the version present in the patch — as Oracle Applications patches are tested to make sure they contain the correct files before they are released.

> **Note:** You can change this behavior by using *options=integrity* on the command line. See Command Line Arguments on page 5-12.

Number of parallel workers

By default, AutoPatch runs database updates and file generation commands in parallel and prompts you for the number of workers. Tasks are assigned to workers, the workers run the tasks to completion, and AutoPatch assigns new tasks.

After you specify the number of workers, AutoPatch displays messages like the following as it begins to update the Oracle Applications products:

```
Performing version checking for driver files...
Copying driver files into installation area...
Determining valid on-site files...
Screening out files not valid for this installation...
Extracting object modules from product libraries...
Performing version checking...
Determining what executables to link...
Determining what Oracle Forms files to generate...
Determining what Oracle Reports libraries to generate...
Determining what Oracle Reports files to generate...
```

> **Note:** AutoPatch runs all database actions based on *phase* order — a grouping of actions in the database portion of the patch that minimizes dependencies. This order is not necessarily the order in which the commands are listed in the database patch driver.

Customized Files

AutoPatch reviews the contents of the applcust.txt file to determine if any registered customized files will be replaced by the patch. If so, it displays a message listing the customized files it will replace.

> **Additional Information:** See Customization Standards in *Oracle Applications Developer's Guide*. See also Monitoring and Controlling Parallel Processes and Managers in Chapter 1.

Preparing for Non-interactive Processing

Non-interactive patching is a way to save time by avoiding some of the prompts and automating the patching process. To use non-interactive patching, you create a defaults file by running AutoPatch interactively using a specific command line option. Then, you tell AutoPatch to run non-interactively by providing the name of the

defaults file plus other associated command line options. After the AutoPatch actions are complete, you perform any post-AutoPatch steps listed in the patch readme file.

> **Additional Information:** See Performing Patching Tasks Non-interactively in *Oracle Applications Maintenance Procedures*.

Messages

AutoPatch generates several types of messages. Each message is recorded in a log file. See Log Files on page 5-7 for a list and description.

Informational Messages

Informational messages are written to the informational message file (adpatch.lgi). This log file uses the same base file name as the main AutoPatch log file, but substitutes a .lgi extension for the .log extension. For example, if the AutoPatch log file is named d123456.log, the AutoPatch informational log file is named d123456.lgi.

For example, AutoPatch writes information pertaining to the files not updated because they are up-to-date in the informational log file.

```
Will not apply POXPOPAA.rdf: Files are identical.
  Patch  : /d01/appl/patch/po/reports/US/POXPOPAA.rdf, v115.3
  On-Site: /d01/appl/115/po/11.5.0/reports/US/POXPOPAA.rdf, v115.3

Not running file 'wip patch/115/sql wipmlprb.pls' against schema 'apptest'
because the corresponding PL/SQL object in the database is up to date.
Revision in File     = 115.8
Revision in Database = 115.8
```

Error messages

When AutoPatch is using parallel processing and an error occurs, it asks if you want to continue. We recommend you do not. After AutoPatch exits, review the main log file (adpatch.log) and the adwork<xxx>.log file to determine the source of the error. If errors exist, resolve the issue and continue. If you cannot resolve the issue, you should:

- Verify that all steps in the readme file were completed.
- Check Oracle*MetaLink* for additional information regarding the patch you are applying.

If the message indicates that a worker has failed its job, you can fix the problem and restart the worker while the manager is running. Some failed jobs are deferred (not immediately reassigned) by the manager. These jobs do not cause the manager or other workers to stop.

> **Additional Information:** See Managing Worker Processes in Chapter 6 of *Oracle Applications Maintenance Procedures.*

Successful completion message

AutoPatch displays messages like the following when processing is complete. If you do not see a completion message, you should investigate the reason why.

```
A job timing report has been generated for the current session. You
should check the file
/d01/appl/115/admin/apptest/out/adt01302.lst
for details.

Purging timing information for prior sessions.
```

```
sqlplus -s APPS/APPS @/d01/appl/115/admin/apptest/ad/11.5.0/admin/sql/adtpurge.sql
10 1000

Done purging timing information for prior sessions.

AutoPatch is complete.

AutoPatch may have written informational messages to the file
/d01/appl/115/admin/apptest/log/adpatch.lgi

You should check the file
/d01/appl/115/admin/apptest/log/adpatch.log for errors
```

The AutoPatch Interface

You run AutoPatch from the command line. It relies on prompts for information, not input screens.

Running AutoPatch

Perform the following steps to start AutoPatch. For a detailed description of all the steps, see Applying a Patch Interactively in Chapter 3 of *Oracle Applications Maintenance Procedures*.

> **Additional Information:** You can add arguments on the command line to refine the way AutoPatch runs. See AutoPatch Modes and Command Line Arguments in this chapter.

Step 1 Set the environment

You must set the environment to apply the configuration parameters that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Unzip patches

Create a patch top directory, if it doesn't already exist. Download the patch and unzip it.

Step 3 Review information in the readme file

This file contains instructions for applying the patch, including any prerequisite patches and/or manual steps.

Step 4 Shut down services

According to the configuration of your system, you may need to shut down concurrent managers, Web server listeners, or forms server listeners.

Step 5 Enable maintenance mode

Use the Change Maintenance Mode option in AD Administration to enable maintenance mode.

Step 6 Start AutoPatch

AutoPatch is located in the AD_TOP/bin directory. However, you start it from the directory that contains the unzipped patch files. Use the following command:

```
$ adpatch
```

Step 7 Respond to prompts

Respond to the AutoPatch prompts for information about your system.

Step 8 Complete the remaining patching tasks

See Interactive Patching in *Oracle Applications Maintenance Procedures* for the remaining tasks required to apply a patch.

Step 9 Exit AutoPatch

When the patching process is complete, AutoPatch displays a message informing you that the process has been completed successfully. If the process did not run to completion, check the log files and determine what caused the problem.

AutoPatch Modes

AutoPatch can apply patches in two specialized modes: test and pre-install. The patch readme file (or other documentation) instructs you when to use one of these modes.

Test Mode

In test mode, AutoPatch doesn't apply the patch. Instead, it lists each file it would have copied, relinked, executed, or generated and shows exactly what actions it would have performed had it applied the patch. It also calls and runs AutoConfig in test mode to determine any impending changes to the configuration files. This means you can see the effects of the patch on your production system before you apply it.

To run AutoPatch in test mode, include *apply=no* on the AutoPatch command line. It works just like applying the patch, except it does not:

- Copy any files from the patch directory to the Oracle Applications file system.
- Archive any object modules into the product libraries.
- Relink any executables.
- Generate any forms, reports, PL/SQL libraries, or menu files.
- Run any SQL or EXEC commands (commands that change the database).
- Instantiate new configuration files.
- Update the patch information files.
- Update patch information and release version in the database.

> **Additional Information:** See Testing a Patch Before Applying It in *Oracle Applications Maintenance Procedures.*

Pre-install Mode

Pre-install mode is generally used during the upgrade process to update AD utilities, apply family consolidated upgrade patches, or work around other patching issues. AutoPatch asks all startup questions except those relating to the database.

> **Note:** Run AutoPatch in pre-install mode *only* if the patch readme instructs you to do so.

To run AutoPatch in pre-install mode, include the *preinstall=y* on the AutoPatch command line. It performs only the following actions.

- Compares version numbers.
- Copies files.
- Relinks FND and AD executables.
- Saves patch information to the file system.

> **Note:** Because AutoPatch does not read driver files in pre-install mode, it copies all product files in the patch to the APPL_TOP directory, even if they should not exist on that node. For example, it will copy forms files to an APPL_TOP that implements only the administration server. Additionally, even if a file in the patch should be both in the APPL_TOP and in another directory (such as in $OA_HTML), AutoPatch copies the file only to the APPL_TOP.

Command Line Arguments

Chapter 1 discusses arguments that are common to various AD utilities. In addition, the following arguments are specific to AutoPatch and can be used to modify and refine its behavior. The Command Line Arguments section in Chapter 1 explains how to enter arguments on the command line. Remember, separate multiple arguments with a blank space. For example, *adpatch apply=n preinstall=y*.

apply	Description
Purpose	Tells AutoPatch whether to run in test mode.
Values	y, meaning that AutoPatch does not run in test mode. n, meaning that AutoPatch does run in test mode.
Default	y
Example	adpatch apply=n

driver	Description
Purpose	Tells AutoPatch the name of the patch driver file. This is usually used during non-interactive processing. It is only valid when the patchtop option is also used.
Values	A driver file name, or comma-separated list of patch driver file names.
Default	None, meaning that AutoPatch prompts for the patch driver file name.
Example	adpatch patchtop=/d1/apps/patches/2344175 driver=c2344175.drv

patchtop	Description
Purpose	Tells AutoPatch the top-level directory for the current patch. This is normally used during non-interactive processing.
Values	A fully qualified directory name.
Default	None, meaning that AutoPatch prompts for the patch directory.
Example	adpatch patchtop=/d1/apps/patches/2344175

preinstall	Description
Purpose	Tells AutoPatch whether to run in pre-install mode. Pre-install mode is used to update AD utilities before an upgrade and to apply family consolidated upgrade patches.
Values	y, meaning that AutoPatch does run in pre-install mode. n, meaning that AutoPatch does not run in pre-install mode.
Default	n
Example	adpatch preinstall=y

uploadph	Description
Purpose	Tells AutoPatch to upload patch history information from the patch history files to the database. AutoPatch exits after uploading the patch history information.
Values	y, meaning that AutoPatch uploads patch history information. n, meaning that AutoPatch does not upload patch history information.
Default	none
Example	adpatch uploadph=y

AutoPatch options

The *options*= argument is used to pass generic options to AutoPatch. It takes the form of a comma-separated list. Enter one option or a comma-separated list of options. For example, *options=nocopyportion,nogenerateportion*. Do not include a space after the comma.

Option	Description
autoconfig	Purpose: Tells AutoPatch to run AutoConfig automatically. Default: autoconfig. Use options=noautoconfig if you are applying a number of patches in sequence and want to run AutoConfig once, after applying the last patch of the sequence. Comments: The more common method is to merge the patches first with AD Merge Patch.
checkfile	Purpose: Tells AutoPatch to either skip running EXEC, SQL, and EXECTIER commands if they are recorded as already run, or to record them as having run after running them. Default: checkfile. Use options=nocheckfile to turn off the checkfile feature. Comments: checkfile provides significant performance benefits.
compiledb	Purpose: Tells AutoPatch to automatically compile invalid objects in the database after running actions normally found in the database driver. Default: compiledb for standard patches. nocompiledb for standard patch translations, documentation patches, and documentation patch translations. Use options=nocompiledb to save time when multiple non-merged patches are applied in a maintenance window. Comments: Merging multiple patches and applying a single merged patch is usually a better strategy.

Option	Description
compilejsp	Purpose: Tells AutoPatch whether to automatically compile out-of-date JSP files. JSP files are only compiled if the patch contains copy actions for at least one JSP file. Default: compilejsp for standard patches. nocompilejsp for standard patch translations, documentation patches, and documentation patch translations. Use options=nocompilejsp to save time when multiple non-merged patches are applied in a maintenance window. Comments: Merging multiple patches and applying a single merged patch is usually a better strategy
copyportion	Purpose: Tells AutoPatch whether to run commands normally found in a copy driver. Default: copyportion. Use options=nocopyportion to tell AutoPatch not to perform copy driver actions. Comments: Useful mostly with unified drivers.
databaseportion	Purpose: Tells AutoPatch whether to run commands normally found in a database driver. Default: databaseportion. Use options=nodatabaseportion to tell AutoPatch not to perform database driver actions. Comments: Useful mostly with unified drivers.
generateportion	Purpose: Tells AutoPatch whether to run commands normally found in a generate driver. Default: generateportion. Use options=nogenerateportion to tell AutoPatch not to perform generate driver actions. Comments: Useful mostly with unified drivers.
hotpatch	Purpose: Tells AutoPatch to apply a patch regardless of whether the Oracle Applications system is in maintenance mode. AutoPatch aborts the patching session if maintenance mode is disabled and the *options=hotpatch* command is not used. Default: nohotpatch.
integrity	Purpose: Tells AutoPatch whether to verify that the version of each file referenced in a copy action matches the version present in the patch. Default: nointegrity Comments: Using options=nointegrity is safe and avoids some AutoPatch overhead.
parallel	Purpose: Tells AutoPatch whether to run actions that update the database in parallel (like sql) and actions that generate files in parallel (like genform). Default: parallel Comments: We do not recommend changing the default, as Oracle Applications patches are tested on systems using parallel processing.

Option	Description
phtofile	Purpose: Tells AutoPatch whether to upload patch history information to the database after applying the patch or to write it to the patch history files in the file system. Default: nophtofile Use options=phtofile to tell AutoPatch not to upload patch history information to the database. Comments: Using phtofile allows you to defer the uploading of patch history information to the database until after the system downtime. Use the adpatch uploadph=y command to upload patch history information from the patch history files to the database during uptime.
prereq	Purpose: Tells AutoPatch whether to check that prerequisite patches have been applied prior to running patch driver files that contain actions normally found in the copy driver. Default: noprereq Use options=prereq to turn on prerequisite patch checking. Comments: options=prereq prevents you from applying a patch without first applying all required prerequisite patches.
validate	Purpose: Tells AutoPatch whether to connect to all registered Oracle Applications schemas at the start of the patch. Default: novalidate. Use options=validate to validate password information for all Oracle Applications schemas. Comments: Useful for finding problems with incorrectly registered Oracle Applications schemas or schemas with invalid passwords.

Restarting AutoPatch

You can stop AutoPatch before it begins processing by typing the *abort* command at any prompt. However, once processing actions have begun, you can stop the actions by shutting down the workers in AD Controller. Or, AutoPatch may quit performing the actions because it encountered an error. You restart patching sessions by typing the start command.

> **Caution:** You stop AutoPatch by typing *abort* at any prompt. However, you cannot abort an AutoPatch session after the process of applying the patch has begun.

If you have shut down the workers, or if AutoPatch quits while performing processing actions, it saves all the actions completed up to that point in restart files. You can decide whether to restart at the point where the processing stopped, or start a new session.

Additional Information: See Restarting a Utility in Chapter 1.

AD Merge Patch

When patches are applied individually, all the AutoPatch tasks — responding to prompts, linking executables, and so on — are performed separately for every patch. AD Merge Patch merges multiple (compatible) patches into a single patch, allowing you to reduce patch application time by eliminating the redundant tasks.

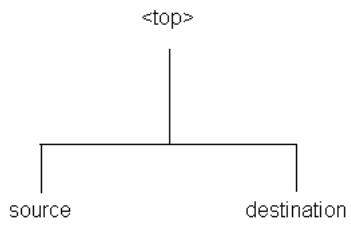
AD Merge Patch can merge patches with split patch driver files (c<patchnum>.drv, d<patchnum>.drv, and g<patchnum>.drv), or unified patch driver files (u<patchnum>.drv). If all the source patches have split driver files, the merged patch has split driver files. If any of the source patches has a unified driver file, the merged patch has a unified driver file.

When merging patches with only unified drivers, AD Merge Patch performs merges based on meta-data and removes duplicate database driver lines. When the source patches contain split drivers files, duplicate database driver line items in the merged patch driver files are skipped during the patching process by the AutoPatch checkfile feature.

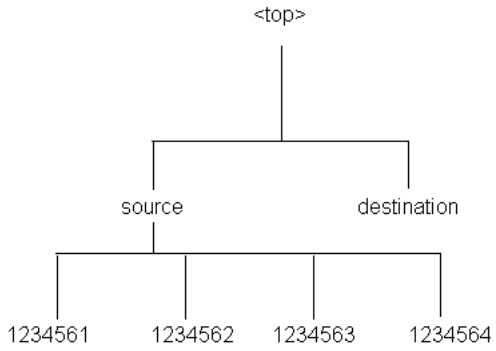
Source and Destination Directories

The *source* directory is where you extract the patches to be merged. The *destination* directory is where the merged patch will be created. AD Merge Patch reads the patch driver files for each patch in the source directory and merges them to create patch driver files in the destination directory. If a file exists in more than one source patch, only the highest revision of the file is copied to the destination directory. The source and the destination directories cannot be child or parent directories of each other.

In setting up the source and destination directories, you should place them under the same parent directory. For example, if the parent directory is named <top>, both the source and destination directories should be subdirectories of <top>.



The source directory must have all patches to be merged as immediate child directories. The patch directories cannot be in a lower directory. For example, a directory structure for merging four patches would look like this:



Naming the Merged Patch

You indicate the name of the merged patch on the command line, using the *-merge_name* option. If you do not provide this information, the patch will be named *merged* by default.

Merging Zipped Patches

Use the *-manifest* option to use a manifest file that contains the name and location of the patch zip files. AD Merge Patch references this file, and unzips the patches listed. It copies the unzipped files into the source directory and includes them, along with any other files in the source directory, in the merged patch.

The manifest file is a text file in which you document the location and names of the patch zip files. The contents of a manifest file will resemble the following:

```
/home/applmgr/patches/p3903945_11i_GENERIC.zip
/home/applmgr/patches/p3892799_11i_GENERIC.zip
/home/applmgr/patches/p3874740_11i_LINUX.zip
```

> **Additional Information:** See Applying a Merged Patch in *Oracle Applications Maintenance Procedures*.

The AD Merge Patch Interface

You run AD Merge Patch and supply the information it needs from the command line. There are no menus are input screens.

Running AD Merge Patch

AD Merge Patch is located in AD_TOP/bin. However, you run it from the parent directory of the source directory.

1. Set the environment.

   You must set the environment to indicate the location of the configuration parameters that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1.

2. Run AD Merge Patch.

   From the <top> directory, run AD Merge patch as follows:

   ```
   admrgpch -s <source directory> -d <destination directory> \
   -merge_name <name> [-manifest <manifest filename>]
   ```

   For example, if you have four patches called 1234561, 1234562, 1234563, and 1234564 located in the source directory /d01/patch_merge/source, and the destination directory is /d01/patch_merge/destination. To create a merged patch named "merge99", you would use the following commands:

   UNIX:

   ```
   $ cd /d01/patch_merge
   $ admrgpch -s /d01/patch_merge/source -d /d01/patch_merge/destination \
     -merge_name merge99
   ```

   Windows:

   ```
   C:\> cd \d01\patch_merge
   C:\> admrgpch -s d:\patch_merge\source -d d:\patch_merge\destination \
        -merge_name merge99
   ```

Patch Wizard

Earlier versions of the Oracle Applications Manager (OAM) introduced the Patch Advisor, a utility used to help determine recommended patches for your system. In

this release, the Patch Advisor features have been enhanced and incorporated into a new utility called Patch Wizard.

With Patch Wizard, you can determine patches that have not been applied to your system. It does not report on all available patches. It compares the patches you have already applied against a list of all recommended Oracle Applications patches. Recommended patches can include high-priority or patches that update to a new code level, such as maintenance packs, family packs, and minipacks.

> **Additional Information:** See Patch Types and Structure in this chapter.

Preparing to Use the Patch Wizard

Because it is web-based, the Patch Wizard does not rely on prompts to gather information about the location of the files it uses, language and platform details, patches you have already applied, and so on. Instead, it provides access to pages where you enter this information using the OAM interface.

Before you submit a request for a patch analysis or download patches, you must access the Patch Wizard main page and prepare to submit the request. This page provides access to various setup tasks, including:

- Set preferences, both site-specific and general, that include the staging directory and various defaults that will apply to the patches you download.
- Set up filters that report only those patches that may affect your system.
- Download the patch information bundle from Oracle*MetaLink*.

The Patch Information Bundle

The list of recommended patches is in a *patch information bundle file* that you download from Oracle*MetaLink* to the staging directory you have previously defined as a preference. This file is updated daily and contains the list of recommended patches as well as metadata for these patches. The Patch Information Bundle metadata contain the readme and LDT file for each recommended patch.

The patch metadata LDT files are FNDLOAD data files included in the top-level directory of all recent patches. The LDT files contain prerequisite patch information and a manifest of all files in the patch with their version numbers. The Patch Information Bundle metadata also include information about the relationships between patches, such as which minipacks are contained in the recommended maintenance pack.

The Patch Wizard loads the Patch Information Bundle data, including LDT files and readme files, into the Oracle Applications database. It uses the metadata to analyze multiple requests. For example, you can narrow the comparison of applied patches to recommended patches to report only on recommended Human Resources patches, or to report only on patches that introduce a new code level.

Concurrent Programs

When you submit a request for patch analysis, Patch Wizard performs and monitors the following tasks using a set of concurrent programs:

- Upload patch information from the Patch Information Bundle to Patch Wizard tables

Patch Wizard loads the Patch Information Bundle metadata, including LDT files and readme files, into the Oracle Applications database.

- Recommend patches based on the current environment and the Patch Information Bundle

  Patch Wizard reports which patches update Oracle Applications at the current code level and which update to a new code level.

- Download patches based on the list of recommended patches

  Patch Wizard can download patches from Oracle*MetaLink* then merge the patches in the Patch Wizard staging directory.

- Analyze lists of patches after downloading them from Oracle*MetaLink*

  You can also use Patch Wizard to upload the metadata for a specific patch or set of patches, and then view information reported from the metadata. For example, you can upload the metadata for a patch, and then view any recommended patches that have not yet been applied and the impact of applying this new patch.

The Patch Wizard Interface

Patch Wizard is a web-based utility in Oracle Applications Manager. Since this utility is accessed through OAM, all the pages share a uniform look and feel.

> **Additional Information:** See OAM Interface in Chapter 1.

Main Page

From this page, you have access to task icons used to set up the Patch Wizard staging directory, manage patch filters, submit concurrent requests, and view recommended patches. In addition, the Results section of this page displays a list of patches based on submitted requests.

Task Icons

From the main page, you access the other Patch Wizard pages by clicking on Task icons. The icons provide links to these pages: Patch Wizard Preferences, Define Patch Filters, Recommended Patches, and Download Patches.

Details Icons

Many Patch Wizard pages allow you to drill down to see more detail. For example, from the Results section of the main page, you can click the Details icon for a specific recommended patch request to view the recommended patch results.

> **Additional Information:** The discussion of each page contains more detail. The OAM help feature also contains information about the Patch Wizard pages.

Accessing Patch Wizard

To access the Patch Wizard, log in to Oracle Applications Manager and choose Patch Wizard from the Site Map.

1. Log in to Oracle Applications Manager.

   Follow the instructions in Accessing OAM in Chapter 1. From the Applications Dashboard, click the Site Map tab.

2. Go to the Patch Wizard main page.

From the Site Map, Patch Wizard is on the Maintenance tab under the Patching and Utilities heading. Click the Patch Wizard link to go to the main page.

3. Set up preferences and filters.

   From the Patch Wizard main page, click the icon in the Tasks column to set preferences and define filters.

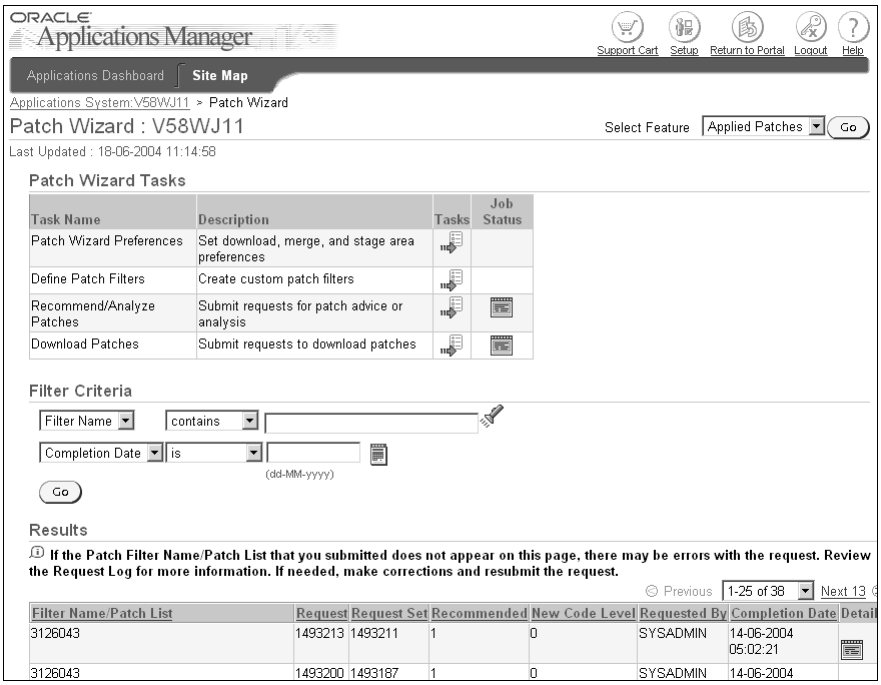
4. Submit request to download the Patch Information Bundle.

   From the main page, click the icon in the Tasks column next to the Recommend/Analyze Patches page. From this page, you can create recommendations, analyze patches, and load the Patch Information Bundle.

   Additional Information: See descriptions of individual pages in this chapter for details. See also Keeping Patches Current in *Oracle Applications Maintenance Procedures*.

Main Page

Use this page to access all features of Patch Wizard and to view the results of your requests for recommended patches.



The Select Feature pull-down menu at the top of the Patch Wizard main page provides access to the Applied Patches, File History and Timing Reports features of OAM. From the Patch Wizard Tasks table in the main Patch Wizard page, you choose the Tasks icons to view:

- The Patch Wizard Preferences page

  From the Preferences page, you set the staging directory, merge patch defaults, the languages and platform defaults for downloading patches, and whether to display or hide hidden patches. You must define the values in the Preferences page before using Patch Wizard.

- The Define Patch Filters page

You usually need to see only those patches that are recommended for your system. Use the Define Patch Filters page to set up filters that report only those patches that may affect your system.

- The Recommend/Analyze Patches page

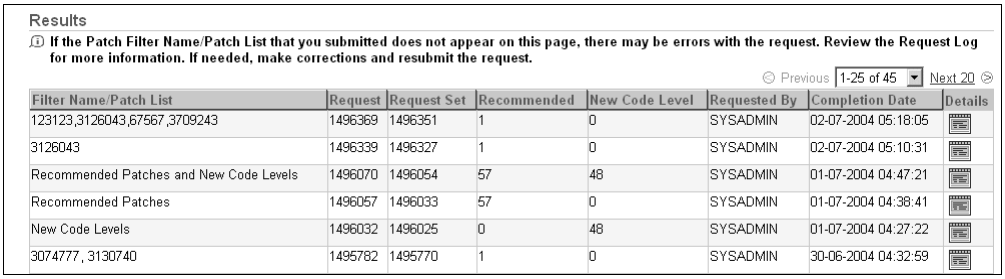
  You can select a filter, then submit the request to run a report of recommended patches based on the filter.

- The Download patches page

  You can download patches, specify the language of patches to download, and merge patches from this page.

Viewing Results

The Results section of the Patch Wizard main page shows the list of all completed recommended patch requests. You can narrow the list of results by entering information in the Filter Criteria section of the main page. For example, you can view only the results of requests submitted on a certain date, or only results that contain a certain text string in the filter name.



Results

If the Patch Filter Name/Patch List that you submitted does not appear on this page, there may be errors with the request. Review the Request Log for more information. If needed, make corrections and resubmit the request.

Previous 1-25 of 45 Next 20

Filter Name/Patch List	Request	Request Set	Recommended	New Code Level	Requested By	Completion Date	Details
123123,3126043,67567,3709243	1496369	1496351	1	0	SYSADMIN	02-07-2004 05:18:05	
3126043	1496339	1496327	1	0	SYSADMIN	02-07-2004 05:10:31	
Recommended Patches and New Code Levels	1496070	1496054	57	48	SYSADMIN	01-07-2004 04:47:21	
Recommended Patches	1496057	1496033	57	0	SYSADMIN	01-07-2004 04:38:41	
New Code Levels	1496032	1496025	0	48	SYSADMIN	01-07-2004 04:27:22	
3074777, 3130740	1495782	1495770	1	0	SYSADMIN	30-06-2004 04:32:59	

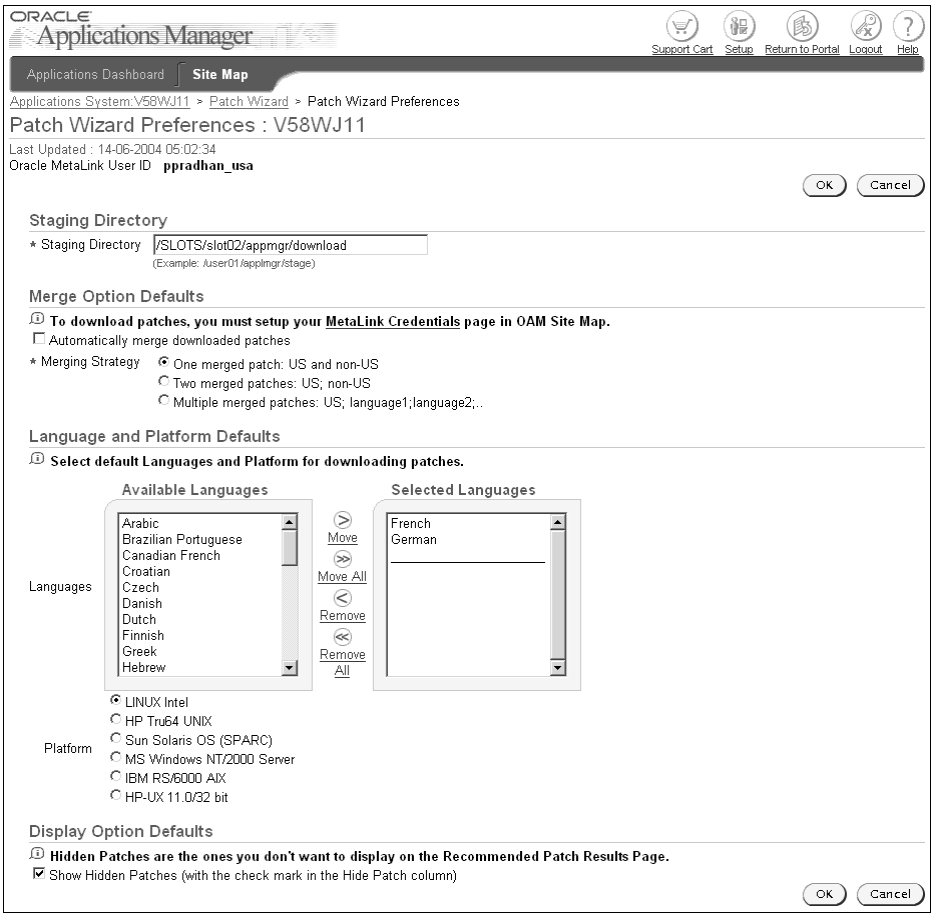
Clicking the icon in the Details column of a specific recommended patch request accesses the Recommended Patch Results page. After setting up and submitting a request, view the details of the recommended patches on this page.

Note that if the patch filter or patch list that you submitted does not appear in the Results section, there may be errors with the request, or the patches may already be applied, or no patches were recommended for the selected criteria. Review the request log for more information. If there are errors, make corrections and resubmit the request.

The information on the Preferences, Define Patch Filters, Recommend/Analyze Patches, Download Patches, and Recommended Patches pages is described in the following sections.

Patch Wizard Preferences

The site-specific information you set on the Patch Wizard Preferences page applies to other functions of the Patch Wizard, such as Recommended Patches and Download Patches. From the Patch Wizard main page, click the Tasks icon to show the Patch Wizard Preferences page.



Staging Directory

The staging directory is where you store files used by Patch Wizard. It is also used by Patch Wizard to create temporary files and subdirectories. These temporary files and directories are deleted after processing.

> **Note:** We recommend you pick a staging directory once and use the same directory each time you run Patch Wizard.

Merge Option Defaults

You can choose to automatically merge patches that you download. You can choose to merge all patches into one merged patch, create two merged patches (one for US patches and one for all non-US patches), or create multiple merged patches (such as one for each separate language).

Language and Platform Details

You can select one or more languages, which are the languages of patches that Patch Wizard will recommend and download. You can also set the platform of the patches you want recommended and downloaded.

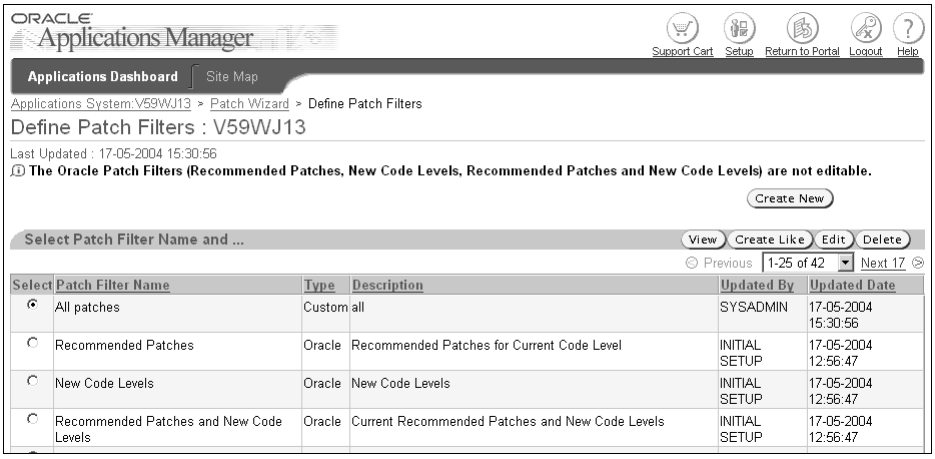
Display Option Defaults

Hidden patches are patches that you choose not to see in your reports. For example, you may be aware of a patch that you plan to download and apply at a later date, but choose not to see it in the current reports. Checking the Show Hidden Patches button on the Preferences screen overrides the hidden patch setting, and all patches, even hidden patches, are reported.

Defining Filters

The Patch Information Bundle file contains information for all recommended patches for all products. If the Patch Wizard were to compare patches in the patch information database against all metadata in this file, the number of recommended patches in the report might be too large to be useful. To avoid this, Patch Wizards provides filters so that only those patch types and products in the metadata that apply to your system are included in the comparison.

From the main page, click the Tasks icon for Define Patch Filters to see all filters created for the current system. Patch Wizard has three pre-seeded filters, and you can create custom filters.



Note that the pre-seeded filters contain "Oracle" in the Type column on this page, and other filters you create contain "Custom" in this column. You cannot edit or delete the three pre-seeded filters, but you can use the three filters as templates to create new filters.

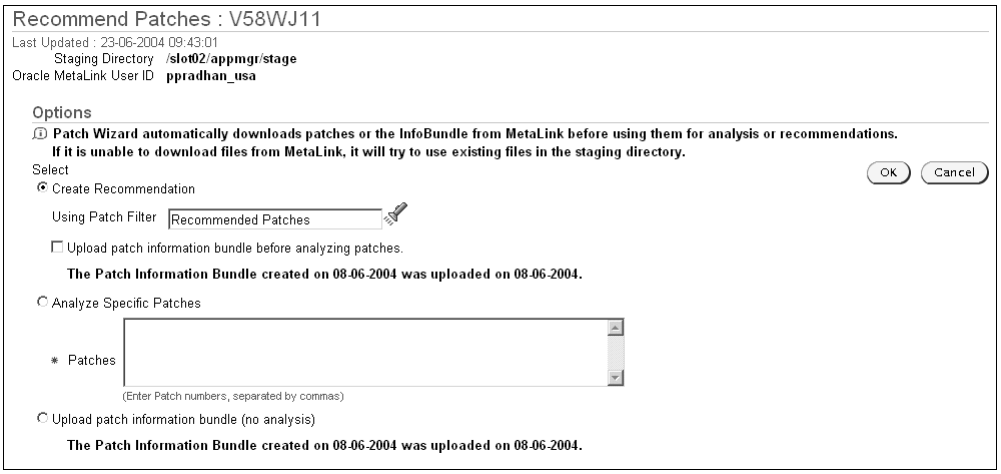
The three pre-seeded filters are New Code Levels, Recommended Patches, and Recommended Patches and New Code Levels. The New Code Levels filter determines recommended patches for minipacks, family packs, and maintenance packs. These new code level patches update versions of a product, family, or the entire Oracle Applications system. The Recommended Patches filter determines recommended patches for the current code level. The Recommended Patches and New Code Levels filter determines recommended patches for both the current and new code levels. Note that you cannot edit the three pre-seeded filters.

From the Define Patch Filters page, click Create New to create a new custom filter. You must enter a unique name and a description for each new custom filter. All licensed product families are listed at the top of the Create New page, and non-licensed product families are listed at the bottom of the page. There are two columns for each product family: Recommended Patches and New Code Levels. Check the appropriate boxes next to each product family to include the patches of each type in the new filter.

If you choose the Create Like button in the Define Patch Filters page, you use an existing filter as a template to create a new filter. The filter criteria from the existing filter is pre-selected in the new filter. Edit the new filter by checking or clearing the boxes, then click Continue to create the new filter.

Recommend Patches

After setting up the Patch Wizard staging area (and optionally creating custom filters), you can submit requests for processing from the Recommend/Analyze Patches page. Click the Tasks icon in the Recommend/Analyze Patches row to access the Recommend Patches page.



The Recommend Patches screen contains the following actions:

- Create Recommendations

  This generates recommendations based on the selected patch filter. You choose one of the three pre-seeded filters or any custom filter you created in the Define Patch Filters page. Patch Wizard uses the filter and compares the patch information database against the metadata patch list to recommend which patches you should apply. Check the "Also import updated Patch Information" box to update the Patch Information Bundle before generating the recommendations.

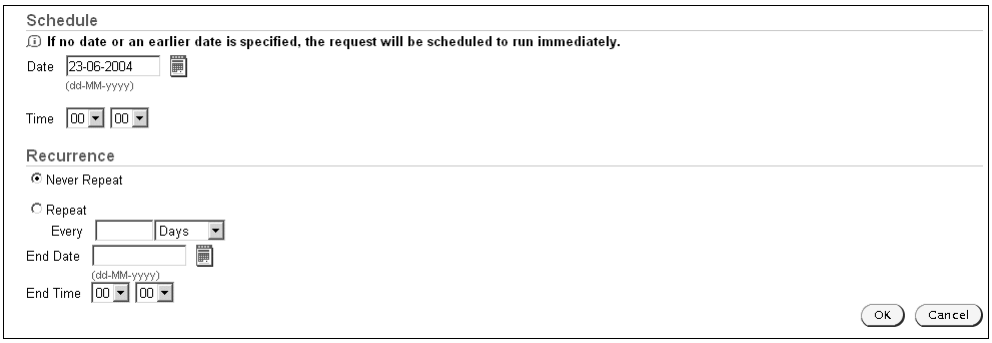
- Analyze Specific Patches

  This generates recommendations for specific patches. After downloading specific patches from Oracle*MetaLink* and placing them in the staging area, you can analyze these patches from this page by entering the patch numbers.

- Upload Patch Information Bundle

  This uploads metadata from the Patch Information Bundle to the Oracle Applications database. If there is no new or updated data in the Patch Information Bundle file, no data will be uploaded to the database.

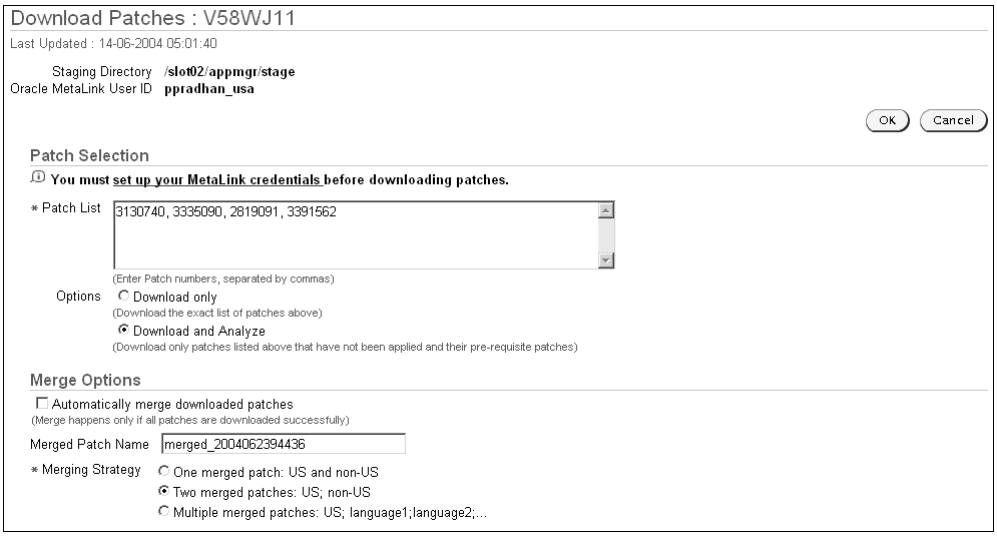
You can enter a date and time in the Schedule section of this page to run the request at a later time. You can also schedule recurring requests by entering the information in the Recurrence section of this page.



Each time you submit a request to upload a patch information bundle, to analyze specific patches, or to recommend patches, the Patch Wizard creates a request ID. The request ID is shown on the Results section of the Patch Wizard Main page.

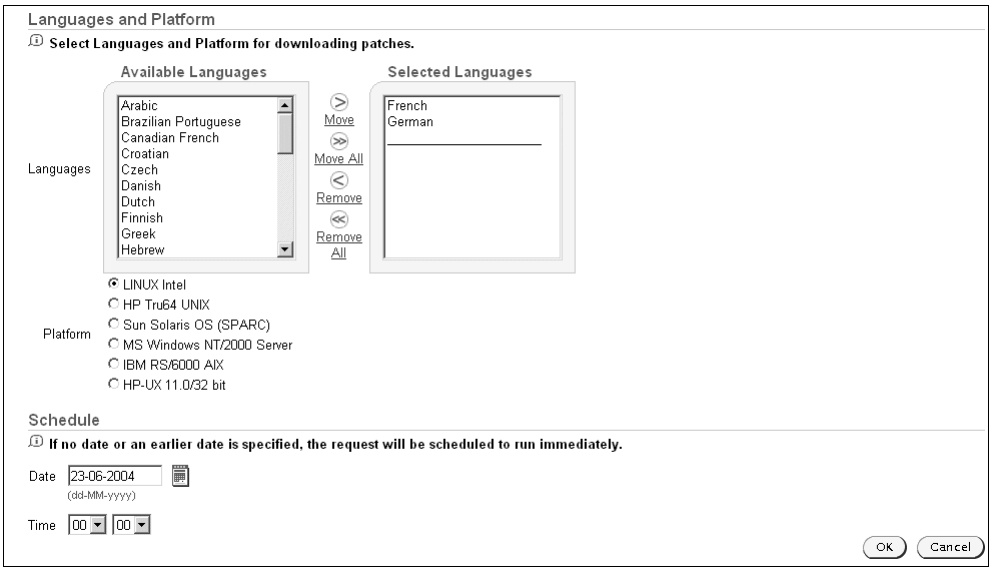
Download Patches

The Download Patches page prompts you for information about the patches to download, then downloads the patches directly from Oracle*MetaLink*. You enter the patch numbers in the input field separated by a comma. You can also choose to analyze the patches while downloading.



The Merge Options section of this page defines how patches should be merged after downloading. The defaults for merging are set on the Patch Wizard Preferences page. If you choose to automatically merge patches while downloading, specify the merged patch name and the merging strategy in this section.

You can select the languages and platform of the patches to download. When you provide information in this section of the page, Patch Wizard only downloads patches that match the selected languages and platform.



You can also provide information in the Schedule section to download at a later time.

Recommended Patches Results

Clicking the Details icon associated with a patch request in the Results section of the Patch Wizard main page accesses the Recommended Patches Results page. This page lists the results of the selected Recommend Patches request that you submitted.

The set of recommended patches are divided in two sections:

- Recommended Patches for Current Code Level
- Patches that introduce New Code Level

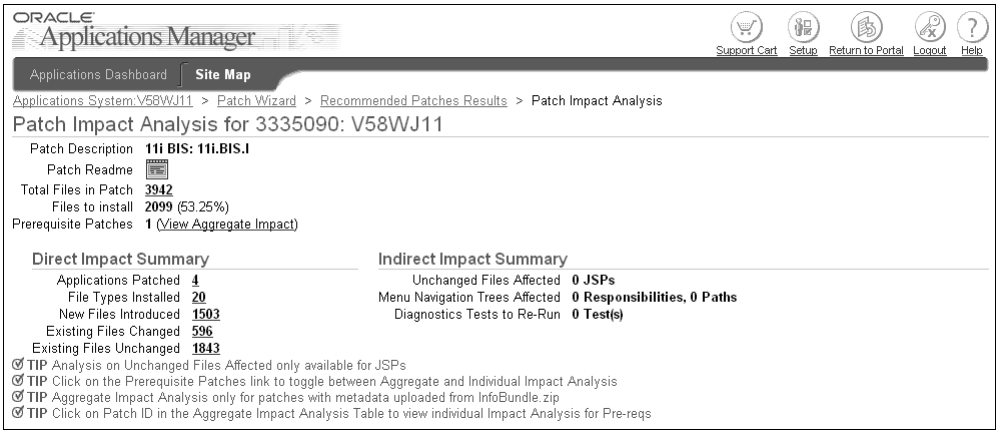
This page has the following columns of information:

- Select: Select this check box and click the Download button if you want the patch number sent to the Download Patches page for submission. You can select any number of patches.
- Patch: This is the patch number of the recommended patch.
- Product Family: The product family to which the patch belongs.
- Product: The product of the patch.
- Patch Description: A brief description of the patch.
- Reason Recommended: The reason the patch is recommended, for example, it is part of a minipack or a family pack.
- Total Pre-Reqs: The number of all prerequisite patches for the recommended patch. If pre-reqs exist, the number will be a hyperlink to a separate Recommended Patches page listing the prerequisite patches.
- Hide Patch: Select this check box to hide the patch from the list of recommended patches. To hide or show selected patches, use the Show Hidden Patches check box at the top of the page and click Redisplay Data to refresh the page. The default values are set in the Patch Wizard Preferences page.
- Impact: Click this icon to access the Patch Impact Summary page. If you submitted a specific patch to analyze, click the Impact icon on the Recommended Patches Request page to view the Patch Impact Analysis Report.

Clicking the Download button carries the selected patch number(s) to the Download Patches page for submission.

Patch Impact Analysis

The Patch Impact Analysis tells which products and files are affected by a particular patch. You can analyze which files are new, which files are changed, and which files are ignored when applying the patch. You can view prerequisite patches required by this patch, and also read the readme file for each patch.



The key information on this page are separated into three sections.

General Patch Information

This section includes the following information:

- Patch Description: A brief description of the patch

- Patch Readme: Click this icon to see the readme file for the patch
- Total Files in Patch: The total number of files in the patch. Click the number link to access the Patch Impact Details page, which lists each file in the patch
- Files to Install: The number of files the patch will install
- Prerequisite Patches: The number of prerequisite patches for the recommended patch

Direct Impact Summary

This section includes the following information:

- Applications Patched: The number of products that will have files updated. Click the number link to access the Patch Impact Details page, which lists each product impacted and how they are impacted.
- File Types Installed: The number of different file types in the patch. Click the number link to access the Patch Impact Details page, which lists the file types and how they impact the system.
- New Files Introduced: The number of new files that will be introduced by the patch. Click the number link to access the Patch Impact Details page, which lists details about each new file introduced.
- Existing Files Changed: The number of existing files in the system that will be changed by the patch. Click the number link to access the Patch Impact Details page, which lists the existing files changed and the new version numbers.
- Existing Files Unchanged: The number of files unchanged because the version in the patch is older than the version in the system. Click the number link to access the Patch Impact Details page, which lists the files in the patch that are the same or earlier versions than those currently in the system.

Indirect Impact Summary

This section includes the following information:

- Unchanged Files Affected: The number of system files with dependencies on patched files
- Menu Navigation Trees Affected: The number of menu navigation trees that will be updated by the patch
- Diagnostics Tests to Re-Run: The number of diagnostic tests to be re-executed after patching

6

Reporting

This chapter describes various reports and views of your system, including information about patches you have applied to your system, statistics for maintenance sessions and the time it takes to run them, and other important system information. It contains the following sections:

- About Oracle Applications Reporting Tools
- Applied Patches Information
- OAM Timing Reports
- AD Job Timing Report
- AD Configuration Report
- AD File Identification Report
- AD Check Digest

About Oracle Applications Reporting Tools

As you use your Oracle Applications, you apply patches and perform other maintenance tasks that modify and enhance your system. Oracle Applications supply numerous reports about system status. For example, you can query the patch history database to see lists of patches applied and files affected, or you can generate a report about the version and translation level of your files. You can also generate reports that contain statistics about how many maintenance sessions are complete, number of jobs in each session, and the time it took to complete the session and individual jobs.

Some of the reporting capabilities are web-based and others are run from the command line.

Web-based Reporting Tools

The following reporting tools are web-based. You access them through the Oracle Applications Manager (OAM).

Applied Patches Information

Oracle Applications tracks all patches applied to your system and stores this history in the Oracle Applications Manager (OAM) database. You can view applied patches information through a set of OAM reports and query options.

OAM Timing Reports

You can view a list of in-progress, stopped, aborted, and completed maintenance sessions, and the times that the sessions were started and when they stopped.

Command Line Reporting Tools

These AD utilities are run from the command line.

AD Job Timing Report

Produced automatically by AutoUpgrade, AutoPatch, and AD Administration to report on long-running processes, this report can also be run manually from the command line to provide summary information about AD utilities sessions.

AD Configuration Report

This report contains information about the installed configuration of Oracle Applications, including product group information, whether Multi-Org or MRC functionality is installed, base language and other installed languages, and so on.

AD File Identification Report

This report identifies the version and translation level of Oracle Applications files.

Applied Patches Information

With the Applied Patches reporting tool, you can view information about the patches applied to your system. This patch history includes information such as:

- Patch number and type (such as minipack and maintenance pack)
- Driver file name and type (copy, database, generate, or unified)
- Platform and version
- APPL_TOP on which the patch was applied
- Contents and language of the patch
- Files changed or copied
- Bug fixes included in each driver file
- Whether the fix was applied successfully, or reason it was not applied
- Timing information (start/end, elapsed time during application, restart time)

> **Note:** If the patch is a maintenance pack, the database also stores the maintenance pack level.

How Patch Information is Stored

AutoPatch stores patch information in the database automatically each time it successfully applies a patch. However, if the patch is not applied successfully, or when you run AutoPatch in pre-install mode, patch history is not written directly to the database, but instead is written to these *patch information files*:

- javaupdates<YYYYMMDDhhmiss>.txt, which contains information about changes to Java files

- adpsv<YYYYMMDDhhmiss>.txt, which contains information about changes to all files except Java files

> **Note:** In the file name, *hh* is in 24-hour format.

Both files are located in the APPL_TOP/admin/<SID> directory. Each time you run AutoPatch, it checks this directory for the existence of the patch information files. If it finds them, it automatically uploads the information they contain to the patch history database. If the upload is successful, AutoPatch then deletes the files. The AutoPatch log file records whether the upload was successful or unsuccessful.

Patch Information and AutoPatch Modes

The way you run AutoPatch affects the way it stores patch history information.

Test mode

When you apply a patch in test mode (using *apply=no* on the command line), AutoPatch does not write to the patch history files, and it does not upload patch history information to the patch history database.

Pre-install mode

When you apply a patch in pre-install mode (using *preinstall=y* on the command line), AutoPatch writes patch history information to the patch information files, and it uploads the contents of these files to the database the next time it runs.

> **Additional Information:** See AutoPatch Modes in Chapter 5.

> **Note:** Running AutoPatch interactively or non-interactively does not affect the way information is stored in the database.

The Applied Patches Interface

The Applied Patches reporting tool is a web-based utility in Oracle Applications Manager. The Simple Search page serves as a home page.

Simple Search page

From this page, you can perform a simple search or access the Advanced Search page. You can use either of these pages to query the database for applied patches (the default) or to see a history of changed files. The results of either type of query appear at the bottom of the search page.

Patch Details page

In the search results for both applied patches or file history, there is a Patch Details column. Clicking any link in this column accesses the Patch Details page. From this page, you can go to the Timing Details page, the Bug Fixes page, or the Action Summary page.

> **Additional Information:** The discussion of each page contains more detail. The OAM help feature also contains information about the Applied Patches utility.

Accessing Applied Patches Information

To query the patch history database for information about patches applied to your system and the files affected, log in to Oracle Applications Manager and choose Applied Patches from the Site Map.

Step 1 Log in to Oracle Applications Manager

Follow the instructions in Accessing OAM in Chapter 1. From the Applications Dashboard, click the Site Map tab.

Step 2 Go the Simple Search page

From the Site Map, Applied Patches is on the Maintenance tab under the Patching and Utilities heading. Click the Applied Patches link to go to the Simple Search page.

Step 3 Select search criteria

From the Simple Search page, you can perform a query for applied patches or file history. Or, you can go to the Advanced Search page to perform a more detailed search.

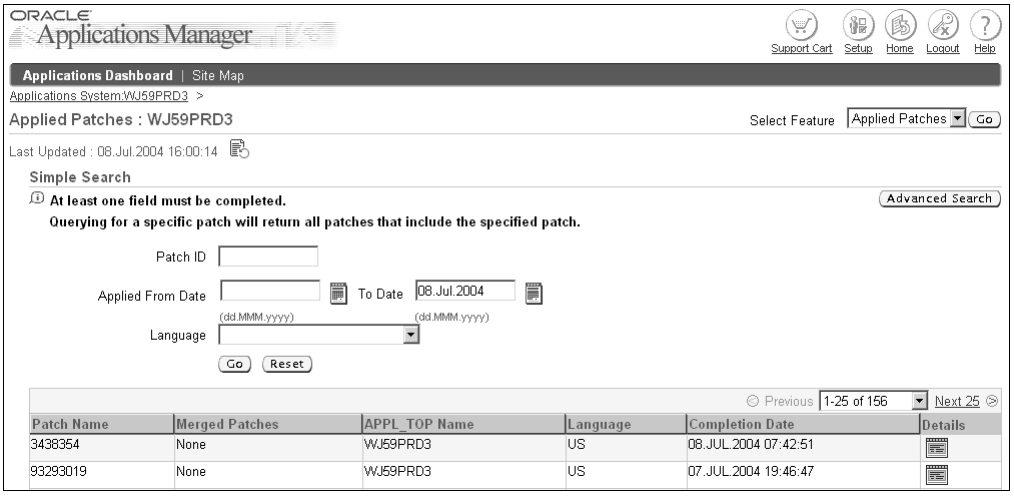
> **Additional Information:** See descriptions of individual pages in this chapter for details. See also Analyzing Applied Patches in *Oracle Applications Maintenance Procedures*.

Applied Patches Search Pages

This section describes queries for applied patches.

Simple Search

You can perform a Simple Search from this page by entering the required information in the input fields.



There are three fields in the Simple Search section:

- Patch ID: Enter the patch number in this field.
- Applied From Date <begin date> To Date <end date>: This field allows you to search for patches that were applied during a specified period of time. Click the calendar icon to select the date or enter the date directly in the field. Some examples for the use of this field are:

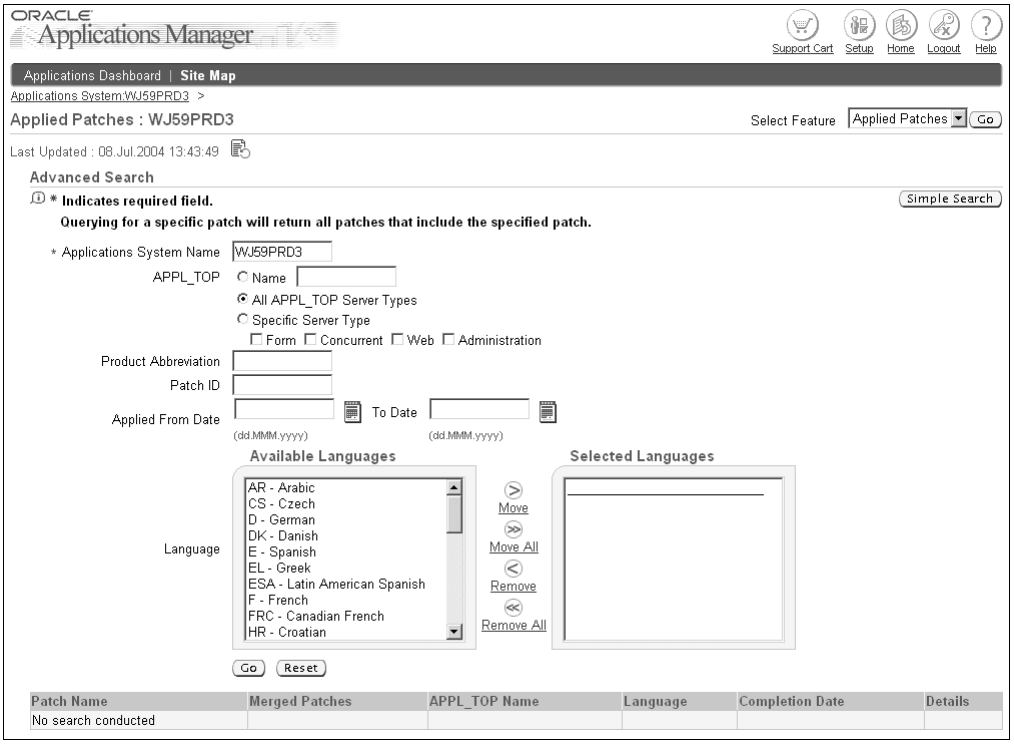
  - Enter only the begin date. This search returns all patches applied from the begin date through today's date.
  - Enter only the end date. This search returns all patches applied up to the end date.
  - Enter the begin date and the end date. This search returns all patches applied between the begin date and the end date.

- Language: This drop down menu allows you to select the language of a patch to be queried. You can select only one language in this field. To select multiple languages, go to the Applied Patches Advanced Search page.

You must enter a value in at least one of the three fields; otherwise an error page appears requesting you to go back and enter a value. To submit the query, click the Go button. The Reset button clears the entered search criteria.

Advanced Search

Click the Advanced Search button to see the Applied Patches Advanced Search page, then enter the search criteria information.



The Advanced Search provides greater granularity of query criteria than the Simple Search page. There are five additional search criteria to narrow the results of a query. These are the available fields in the Advanced Search page.

- Applications System Name (required): Defaults to the name of your Oracle Applications system. If you have migrated applied patches information from another system, and want to search those records, enter the name of that system.
- APPL_TOP: Select Name and enter the name of the APPL_TOP where the patches were applied or select the type of server where the patches were applied. The server type options are All APPL_TOP Server Types or Specific Server Type. The server types are Forms server, Concurrent Processing server, Web server, and

Administration server. When selecting Specific Server Type, choose one server or multiple servers by checking the appropriate check boxes.

- Product: Enter the product short name of the product that owns the patch in this field. For example, "ad" or "inv". The product short names for gl, ap and fa are SQLGL, SQLAP and OFA respectively. For all other products, the short name is the uppercase equivalent to the product abbreviation. This field is not case sensitive.
- Patch ID: Enter the patch number in this field.
- Applied From Date <begin date> To Date <end date>: This field allows you to search for patches that were applied during a specified period of time. Click the calendar icon to select the date or enter the date directly in the field.
- Language: This list box allows you to select the language of a patch to be queried. You can select one language or multiple languages by selecting a language in the Available Languages box and clicking the Move button.

Search Results

After a search, the results appear at the bottom of the search page. The report shows the result of either a Simple Search or Advanced Search. If the results section contains multiple pages of retrieved information, use the Previous and Next links or the drop down list to navigate from page to page. The retrieved patch information is presented in increments of 25 line items per page. Each line item represents an applied patch.

The details provided for each patch are:

- Patch Name: This is the name of the patch.
- Merged Patches: Shows a list of merged patches.
- APPL_TOP Name: This is the name of the APPL_TOP where the patches were applied.
- Language: Indicates the language of the patch.
- Completion Date: This is the date and time the patch was applied and completed.
- Details: Clicking this link accesses the Patch Details report.

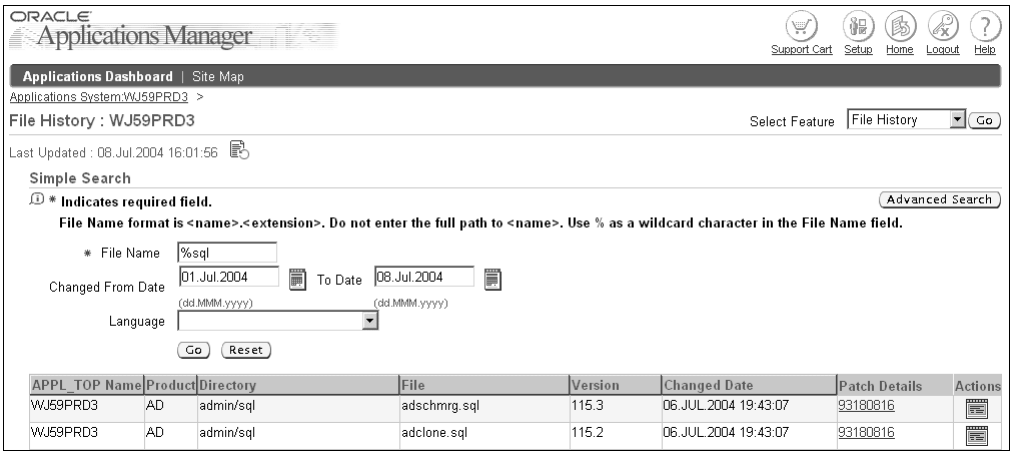
Clicking on a Details icon in the report opens the Patch Details report. This report provides details for a specific patch as explained in the OAM Help system. From the Patch Details report, you can drill down further and access reports showing timing details for the patch, a report of all files copied to the file system by this patch, all bug fixes that were applied by this patch, and a report of all actions taken by a driver when applying this patch.

File History Search Pages

To search for files that have been updated by a patch, click the File History option in the Select Feature drop-down menu on the Applied Patches search pages.

Simple Search

You can perform a Simple Search from this page by entering the required information in the input fields.



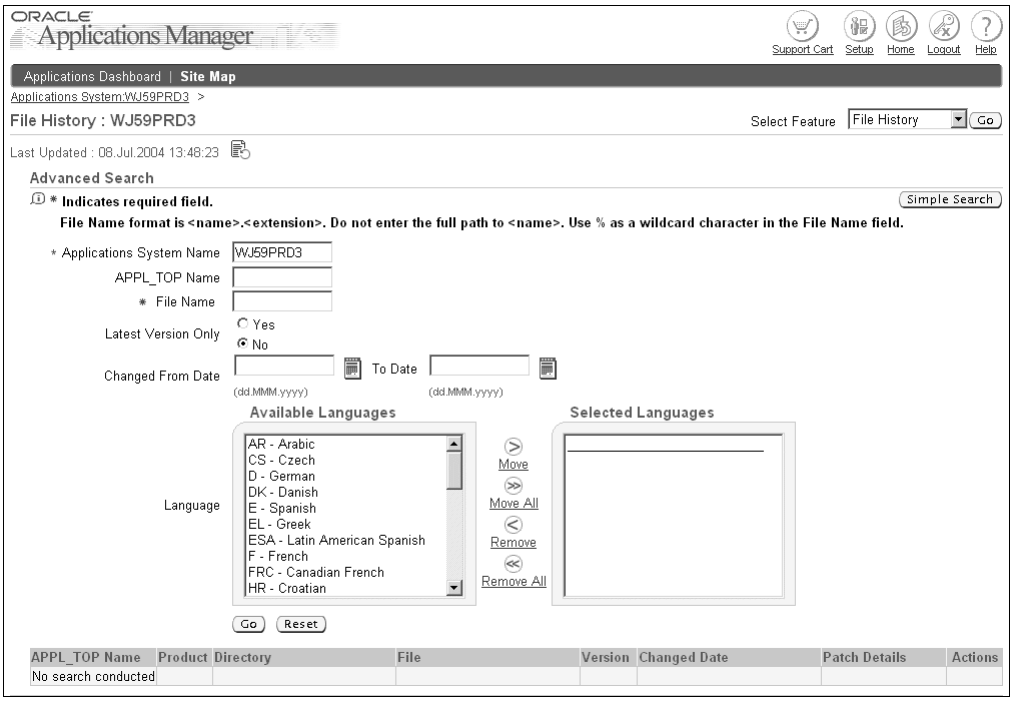
There are three fields in the Simple Search section:

- File name: Enter the name of a file in this field. Do not include a directory path. This field is required and is case-sensitive. You can use the % wildcard symbol in combination with literal characters.
- Changed From Date <begin date> To Date <end date>: This field allows you to search for files that were updated during a specified period of time. Click the calendar icon to select the date or enter the date directly in the field. Some examples for the use of this field are.
  - Enter only the begin date. This search returns file history information from the begin date through today's date.
  - Enter only the end date. This search returns file history information up to the end date.
  - Enter the begin date and the end date. This search returns file history information between the begin date and the end date.
- Language: This drop down menu allows you to select the language of a file to be queried. You can select only one language in this field. To select multiple languages, go to the File History Advanced Search page.

You must enter a value in the File Name field, otherwise, a window appears requesting you to go back and enter a value. To submit the query, click the Go button. The Reset button clears the entered search criteria.

Advanced Search

Click the Advanced Search button to see the Advanced Search page, then enter the search criteria information.



There are four search criteria, in addition to the three found on the Simple Search page, to narrow the results of a query. The available fields in the Advanced Search page are:

- Applications System Name (required): Defaults to the name of your Oracle Applications system. If you have migrated file history information from another system, and want to search those records, enter the name of that system.
- APPL_TOP name: This is the name of the APPL_TOP containing the file.
- File name (required): Enter the name of a file in this field. Do not include a directory path. This field is required and is case-sensitive. You can use the % wildcard symbol in combination with literal characters.
- Latest Version Only: The options are Yes or No. Yes returns information for only the latest version of the file. No returns information for all versions of the selected file.
- Changed From Date<begin date> To Date <end date>: This field allows you to search for file history information spanning a specified period of time. Click the calendar icon to select the date or enter the date directly in the field.
- Language: This list box allows you to select the language of a patch to be queried. You can select one language or multiple languages by selecting a language in the Available Languages box and clicking the Move button.

Search Results

After a search, the results appear at the bottom of the page. Each line item represents the changing of a file due to its inclusion in a patch. The details provided for a file are:

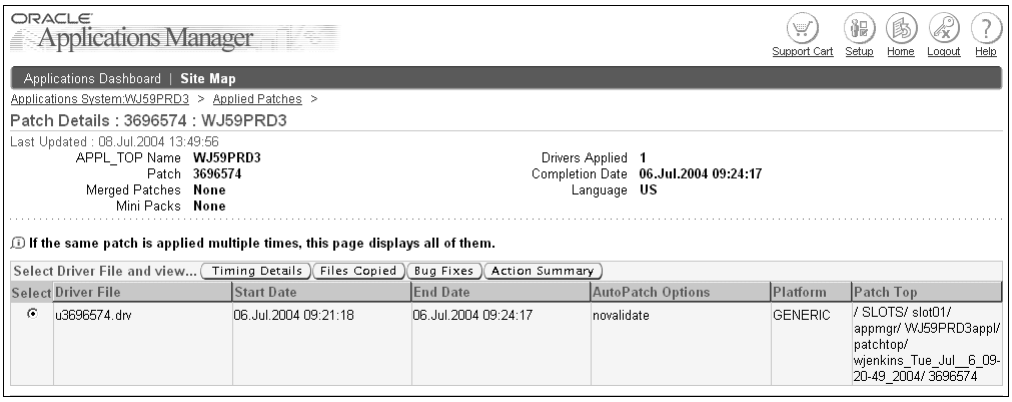
- APPL_TOP Name: This is the name of the APPL_TOP containing the files.
- Product: This is the product for the product that owns the file.
- Directory: This is the directory path where the file is located.
- File: This is the name of the file.

- Version: This is the version number of the file.
- Changed Date: This is the date this version of the file was updated by a patch.
- Patch Details: Click on the patch number to see the Patch Details report for the patch in which the file was included.
- Action Summary: Click on the icon to see the Action Summary report for the action that updated the file.

If a file has never been patched, "The above criteria resulted in no rows" appears in the APPL_TOP Name column.

Patch Details

Clicking the Details link in a selected row from the results section of the Applied Patches page opens the Patch Details report. The Patch Details report provides details for a specific patch. The patch summary information is carried over from the Results section of the Applied Patches Search page and appears at the top of the Patch Details report.



This report contains the following information:

- Select: This option button determines which driver file details are presented in the Timing Details report, Files Copied report, the Bug Fixes report, or the Action Summary report.
- Driver File: This is the name of the driver file.
- Start Date: This is the date and time the application of the driver file began.
- End Date: This is the date and time the application of the driver file completed.
- AutoPatch Options: This column displays any command line options used to run the driver file.
- Platform: This is the platform of the driver file.
- Patch Top: This is the location of the driver when it was run.

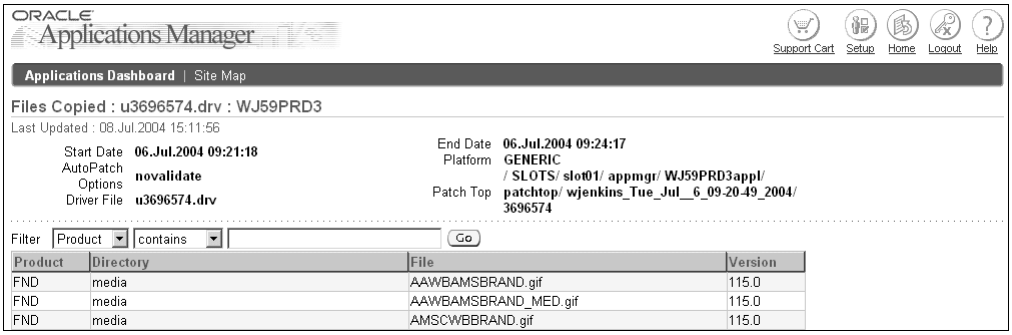
To see additional details for a patch, click one of the following buttons on the report:

- Timing Details: Clicking this button takes you to the AutoPatch Timing Details report.
- Files Copied: Clicking this button takes you to the Files Copied report.
- Bug Fixes: Clicking this button takes you to the Bug Fixes report.
- Action Summary: Clicking this button takes you to the Action Summary report.

Note: The AutoPatch Timing Details report can also be accessed through the Timing Reports link from the Maintenance tab on the OAM Site Map.

Files Copied

The Files Copied report lists all files copied to the file system as a result of the actions in the selected driver file. You access this report by selecting a driver file in the Patch Details report and clicking the Files Copied button.



This report provides the following information about the files copied:

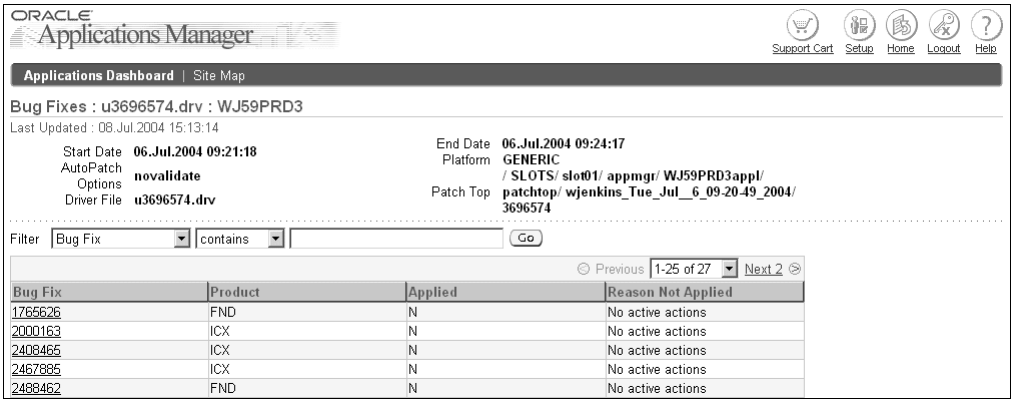
- Product: This is the product short name for the product that owns the file.
- Directory: This is the directory path where the file was copied.
- File: This is the name of the file.
- Version: This is the version number of the copied file.

You can sort each of these columns by clicking the column title at the top of the report. If there are no files copied in the patch, no rows are displayed.

If the number of files copied exceeds 1,000, the report lists only the first 1,000 files. Use the filter to reduce the number of files in the report. Click the Total Count button to see the total number of files copied by the patch

Bug Fixes

The Bug Fixes report lists all bug fixes included in the selected driver file. Each line item represents a bug fix. You access this report by selecting a driver file in the Patch Details report and clicking the Bug Fixes button.



This report provides the following information about bug fixes:

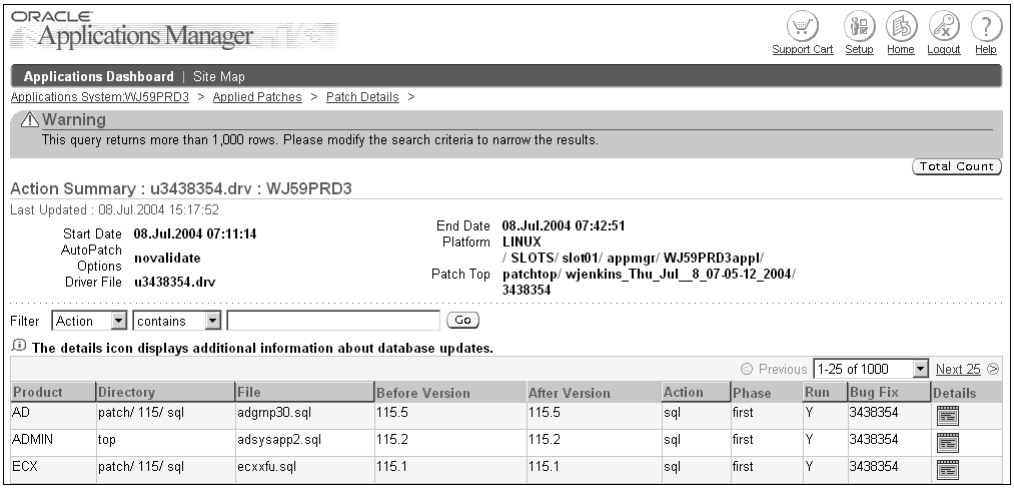
- Bug Fix: This is the bug number of the bug fixed as a result of the selected driver file. The items in this column are links. Clicking an item accesses the Action Summary report.
- Product: This is the product short name for the product whose bug was fixed.
- Applied: This represents whether the bug fix was applied.
- Reason Not Applied: If the bug fix was not applied, the reason is stated here.

You can sort each of these columns by clicking the column title at the top of the report. If there are no bug fixes in the patch, no rows are displayed.

If the number of bug fixes exceeds 1,000, the report lists only the first 1,000. Use the filter to reduce the number of items in the report. Click the Total Count button to see the total number of bug fixes in the patch

Action Summary

The Action Summary report provides summary information for the actions of a selected driver file. Each line item represents a performed action. You access this report either by selecting a driver file in the Patch Details report and clicking the Action Summary button or by clicking a bug fix number in the Bug Fix column of the Bug Fixes report.



The Action Summary report provides the following summary information:

- Product: This is the product short name for the product that owns the file referenced by the action.
- Directory: This is the directory path for the file referenced by the action.
- File: This is the name of the file referenced by the action.
- Before Version This is the version of the file before the update.
- After Version: This is the version of the file after the update.
- Action: This is the type of action performed on the updated file.
- Phase: This is the phase in which the action occurred.
- Run: This signifies whether the action was executed.

- Bug Fix: This is the bug number of the bug fixed as a result of the selected driver file.
- Details: This link is active if AutoPatch performed database actions. Click this link to access the Action Details report.

You can sort each of these columns by clicking the column title at the top of the report. If the number of actions exceeds 1,000, the report lists only the first 1,000. Use the filter to reduce the number of items in the report. Click the Total Count button to see the total number of actions in the patch.

Action Details

The Action Detail report is accessed by clicking the details link in a selected row of the Action Summary report. The Action Summary information is carried over and presented at the top of the report.

The action detail provided are:

- Arguments: This is the specific arguments for SQL and EXEC commands.
- Command Modifier: This is the SQL or EXEC command modifier in the database driver.
- Check Object: This is the name of the database object to check for, along with name and password of the schema where AutoPatch looks for the check object. (This is "none none none" for most SQL commands and is not specified for EXEC commands.)
- Elapsed Time: This is the time required to complete the action.
- Start Time: This is the date and time the action began.
- Restart Time: This is the date and time the action was restarted.
- End Time: This is the date and time the action completed.
- Restarted?: States whether the action was restarted.

N/A in the report represents action details that are not specified. For example, in the Arguments field, N/A means no additional arguments were specified.

OAM Timing Reports

Timing Reports is an Oracle Applications Manager (OAM) utility that lists statistics about AutoPatch and AD Administration maintenance sessions that run parallel workers. As they run processing sessions, both AutoPatch and AD Administration store information about the session in database tables. You can access this information, either during the session or after it is complete, through the OAM interface.

During a parallel session, AD utilities assign processing jobs to workers. For jobs that affect the database, job actions are grouped in phases, which reduces dependencies

between jobs — workers don't have to wait for another worker to complete a dependent job before completing their assigned task.

> **Additional Information:** See Using Parallel Processing in Chapter 1.

The Timing Reports utility lists processing tasks and provide details about the elapsed time for phases, jobs, and sessions. The information includes:

- Jobs run successfully on the first try
- Failed jobs that were restarted and then run successfully
- Failed jobs that were skipped
- Long-running jobs
- Summary information for each parallel phase
- How long it took to run a job
- Overall elapsed time for each session

The OAM Timing Reports Interface

The Timing Reports consists of a Main page and a Timing Details page, which provides links to reports about specific maintenance session information.

Main Page

From the Timing Reports main page, you can view a list of all in-progress, stopped, aborted and completed maintenance sessions. From this page, you can click the Details icon to access the Timing Details page.

Timing Details Page

There are two types of Timing Details reports — those associated with an AutoPatch session and those associated with an AD Administration session.

> **Additional Information:** The discussion of each page contains more detail. The OAM help feature also contains information about the Timing Reports.

Accessing OAM Timing Reports

To access the Timing Reports main page, log in to Oracle Applications Manager and choose Timing Reports from the Site Map.

Step 1 Log in to Oracle Applications Manager

Follow the instructions in Accessing OAM in Chapter 1. From the Applications Dashboard, click the Site Map tab.

Step 2 Access Timing Reports

From the Site Map, Timing Reports is on the Maintenance tab under the Patching and Utilities heading. Click the Timing Reports link to go to the Main page.

Step 3 Filter the results

There is a filter at the top of the page that allows you to narrow the contents of the list. You can filter based on the following status of the tasks: All tasks, In-progress tasks,

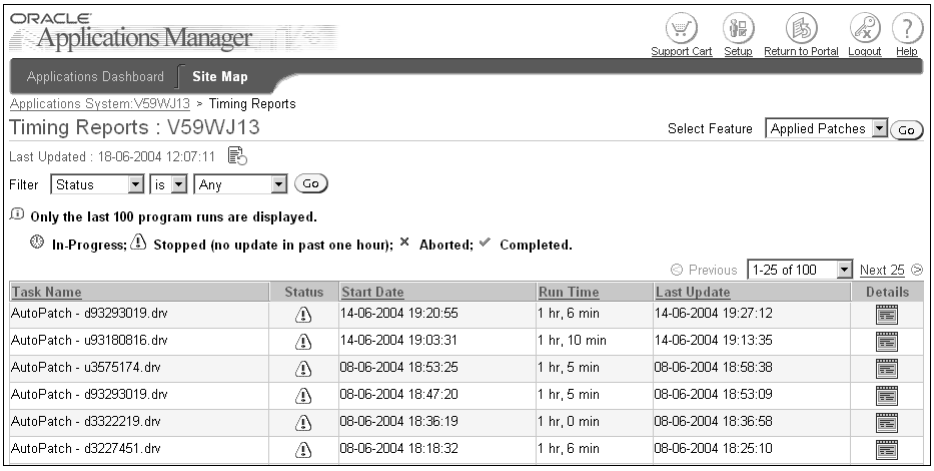
Stopped tasks, Aborted tasks, or Completed tasks. You can also filter by Task Name, Start Date and Run Time. Click Go to activate the filter.

> **Additional Information:** See descriptions of individual pages in this chapter for details. See also AD Administration and AutoPatch Timing Information in *Oracle Applications Maintenance Procedures*.

Timing Reports Main Page

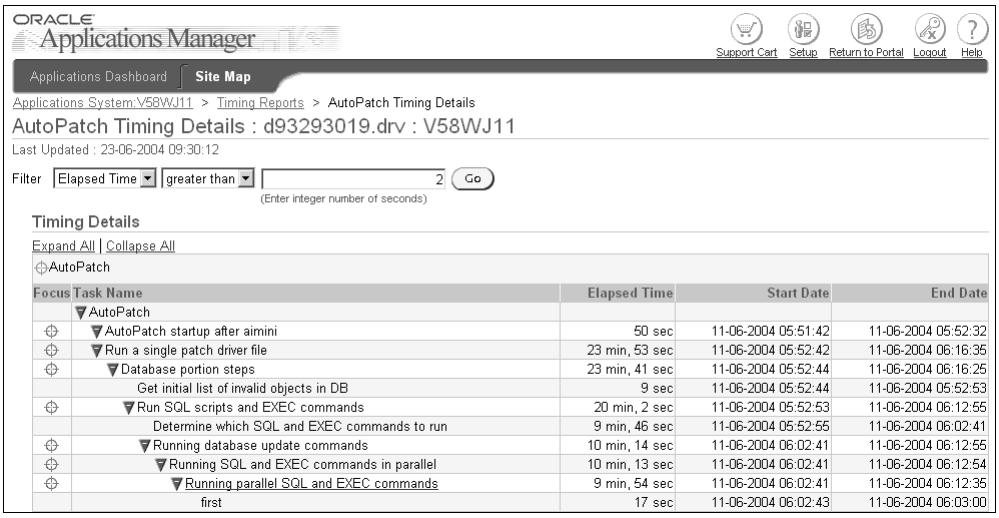
The Timing Reports main page contains the following information for each maintenance session:

- Task Name: This is the name and brief description of the maintenance session.
- Status: This is the status of the timing report. A clock icon means the session is still in-progress, an exclamation icon means the session has stopped, an X icon means the session was aborted (that is, the AD utility was restarted with the restart=no option), and a check mark means the session has completed.
- Start Date: This is the date and time the maintenance session began.
- Run Time: This is the time required to complete the maintenance session.
- Last Update: This is the time the timing information was last updated.
- Details: By clicking the Details icon, you can access the Timing Details for the maintenance session.



AutoPatch Timing Details

Clicking the Details link of a selected row with an AutoPatch Task Name in the Timing Reports list opens the AutoPatch Timing Details report. The AutoPatch Timing Details report provides details for a specific session of AutoPatch.



The AutoPatch Timing Details report lists every task performed in the maintenance session. The Timing Details section contains the following information for each task:

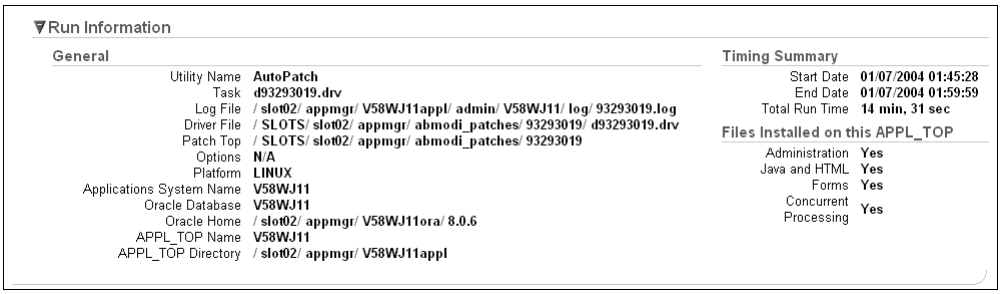
- Focus: Select the circle icon next to a task to see just the sub-tasks within it.
- Task Name: This is the name of the task. Click the blue triangle icon to expand or contract the sub-tasks within the task. The underlined Task Names are links to the Job Timing report for that particular task.
- Elapsed Time: This is the time required to complete the task. This field is not applicable for stopped or in-progress tasks.
- Start Date: This is the date and time the task began.
- End Date: This is the date and time the task completed. This field is not applicable for stopped or in-progress tasks.
- Number of jobs in this task: This is the total number of jobs in the task. This field appears for stopped or in-progress tasks only.
- Number of jobs completed: This is the number of jobs completed. This field appears for stopped or in-progress tasks only.

The filter at the top allows you to adjust the list of tasks based on the elapsed time of tasks. The default list shows all tasks with elapsed time of greater than 4 seconds. Use the Expand All link to see all sub-tasks and the Collapse All to see just the top level task.

When you access the AutoPatch Timing Details report for a stopped or in-progress task, the page defaults to display the most recently performed sub-tasks. For in-progress tasks, you can use the Refresh icon to get the latest running tasks. The Refresh icon is a picture of a page with a green circular arrow.

Run Information

Additional AutoPatch task information is available by clicking the + icon for the Run Information section at the bottom of the AutoPatch Timing Details page. The subsections in Run Information are General, Timing Summary and Files Installed on this APPL_TOP.



General This subsection contains the following information:

- Utility Name: This is the utility used to perform the task.
- Task: This is the task performed.
- Log File: The name and location of the log file.
- Driver File: This is the name and location of the patch driver file.
- Patch Top: This is the location of the patch driver files.
- Options: This is the command options used when running AutoPatch.
- Platform: This is the platform of the system.
- Applications System Name: This is the name of the Applications system on which the task was performed.
- Oracle Database: This is the name of the database.
- Oracle Home: This is the path to the Oracle home used to link the executables.
- APPL_TOP Name: This is the name of the APPL_TOP.
- APPL_TOP Directory: This is the APPL_TOP path.

Timing Summary This subsection contains the following information:

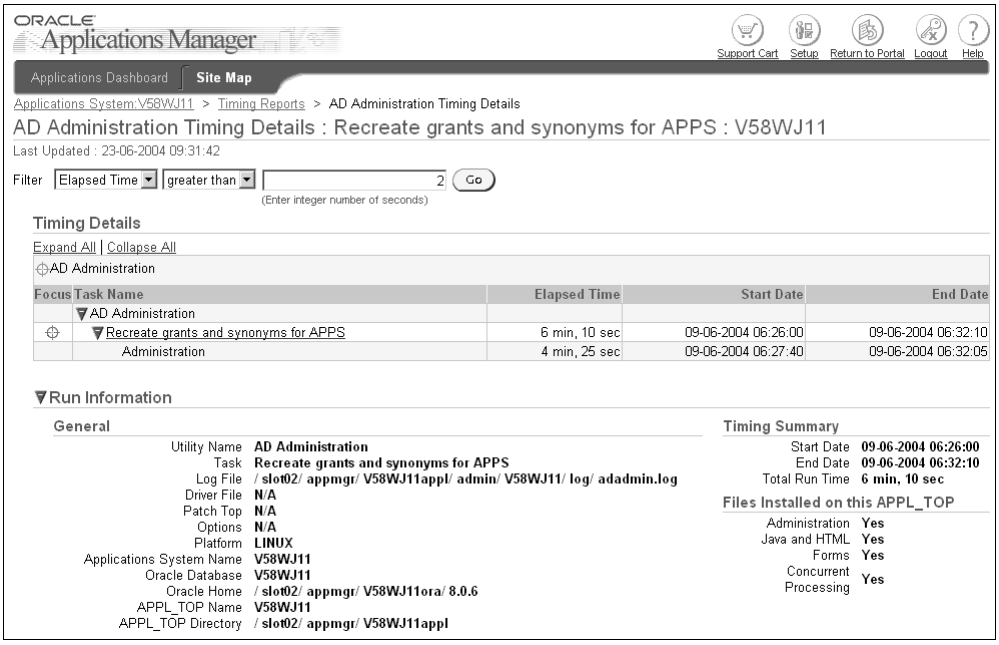
- Utility Start Date: This is the date and time the task began.
- End Date: This is the date and time the task completed. This field is not applicable for stopped or in-progress tasks.
- Total Run Time: This is the time required to complete the task. This field is not applicable for stopped or in-progress tasks.

Files Installed on this APPL_TOP This subsection contains the following information:

- Administration: States whether the APPL_TOP on which the task was performed is an administration server.
- Java and HTML: States whether the APPL_TOP on which the task was performed is a web server.
- Forms: States whether the APPL_TOP on which the task was performed is a forms server.
- Concurrent Processing: States whether the APPL_TOP on which the task was performed is a concurrent processing server.

AD Administration Timing Details

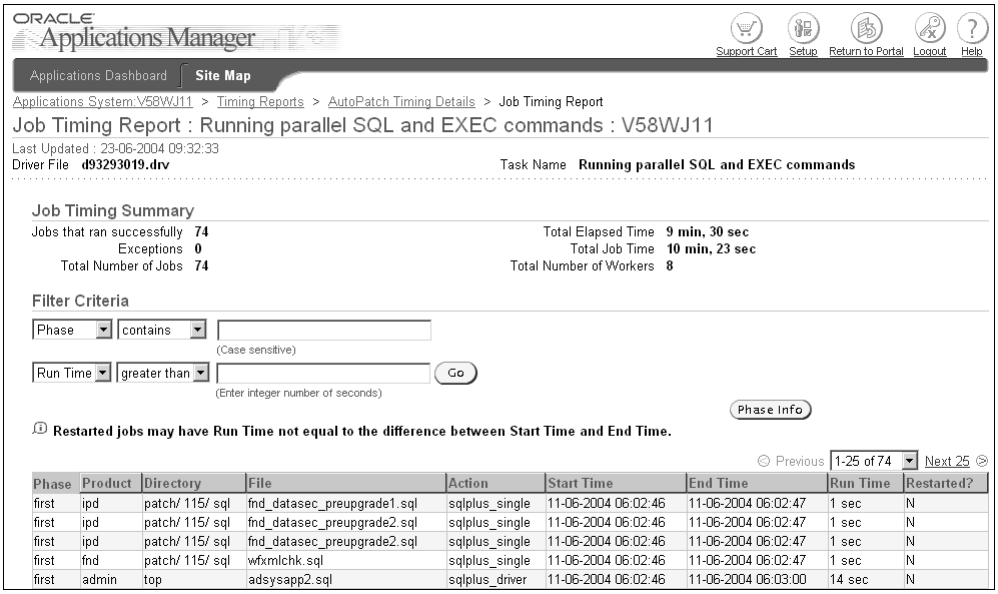
Clicking the Details link of a selected row with an AD Administration Task Name in the Timing Reports list opens the AD Administration Timing Details report. The AD Administration Timing Details report provides details for a specific session of AD Administration.



The Timing Details and Run Information sections contain the same types of information for each task as the AutoPatch Timing Details report.

Job Timing

The underlined Task Names in the AutoPatch Timing Details report and the AD Administration Timing Details report are links to the Job Timing report for that particular task. The Job Timing report provides timing information for each job within the selected task.



The Job Timing Summary information appears at the top of the Job Timing report and the details appear at the bottom.

The Job Timing Summary information includes:

- Jobs that ran successfully: This is the number of successful jobs.
- Exceptions: This is the number of jobs that did not complete successfully. If exceptions exist, this will be a hyperlink to the Exception report.
- Total Number of Jobs: This is the number of jobs within the task.
- Total Elapsed Time: This is the time required to complete the task.
- Total Job Time: This is the time required to complete the jobs within the task.
- Total Number of Workers: This is the number of workers used to perform the task.

The Job Timing Details section contains the following information for each job:

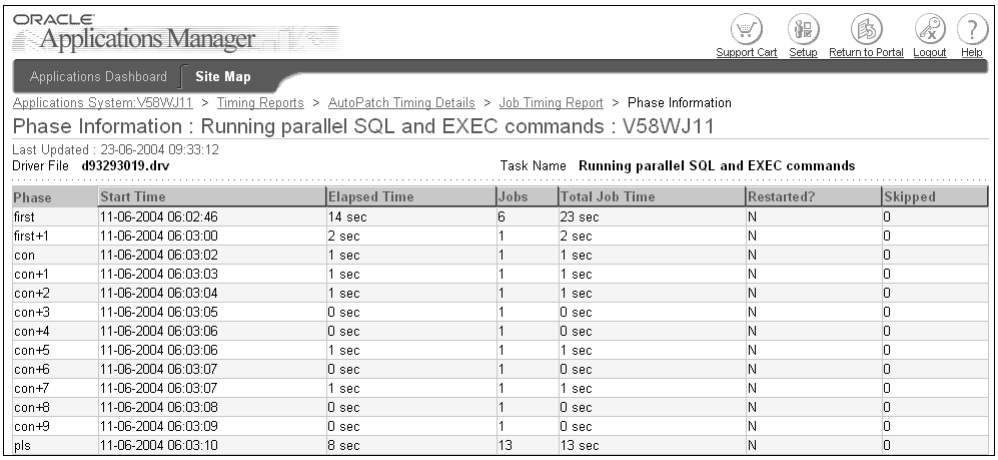
- Phase: This is the database processing phase.
- Product: This is the product abbreviation for the product being updated.
- Directory: This is the directory path of the file run by the job.
- File: This is the file used to perform the job.
- Action: This is the action of the job.
- Start Time: This is the date and time the job began.
- End Time: This is the date and time the job completed.
- Run Time: This is the total time of the job.
- Restarted?: States whether the job was restarted

The filters at the top of the Details section allow you to adjust the list of jobs based on the property and run time of jobs. You can filter based on the following properties of the jobs: Phase, Product, Directory, File, Action, or Restarted. Click Go to activate the filter.

Clicking the Phase Info button opens the Phase Information report.

Phase Information

The Phase Information report provides timing information by phase for a task selected in either the AutoPatch Timing Details report or AD Administration Timing Details report.


ORACLE Applications Manager

Support Cart | Setup | Return to Portal | Logout | Help

Applications Dashboard | Site Map

Applications System:V58WJ11 > Timing Reports > AutoPatch Timing Details > Job Timing Report > Phase Information

Phase Information : Running parallel SQL and EXEC commands : V58WJ11

Last Updated : 23-06-2004 09:33:12

Driver File **d93293019.drv** Task Name **Running parallel SQL and EXEC commands**

Phase	Start Time	Elapsed Time	Jobs	Total Job Time	Restarted?	Skipped
first	11-06-2004 06:02:46	14 sec	6	23 sec	N	0
first+1	11-06-2004 06:03:00	2 sec	1	2 sec	N	0
con	11-06-2004 06:03:02	1 sec	1	1 sec	N	0
con+1	11-06-2004 06:03:03	1 sec	1	1 sec	N	0
con+2	11-06-2004 06:03:04	1 sec	1	1 sec	N	0
con+3	11-06-2004 06:03:05	0 sec	1	0 sec	N	0
con+4	11-06-2004 06:03:06	0 sec	1	0 sec	N	0
con+5	11-06-2004 06:03:06	1 sec	1	1 sec	N	0
con+6	11-06-2004 06:03:07	0 sec	1	0 sec	N	0
con+7	11-06-2004 06:03:07	1 sec	1	1 sec	N	0
con+8	11-06-2004 06:03:08	0 sec	1	0 sec	N	0
con+9	11-06-2004 06:03:09	0 sec	1	0 sec	N	0
pls	11-06-2004 06:03:10	8 sec	13	13 sec	N	0

The general information presented at the top of the Phase Information report are:

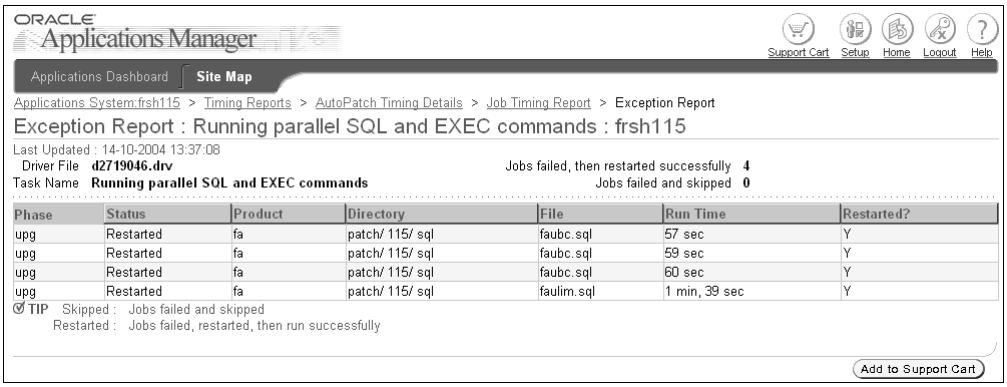
- Driver File: This is the name of the driver file.
- Task Name: This is the name of the task performed.

The Phase Information details include:

- Phase: This is the database processing phase.
- Start Time: This is the date and time the phase began.
- Elapsed Time: This is the time required to complete the phase.
- Jobs: This is the number of jobs in the phase.
- Total Job Time: This is the time required to complete the jobs within the phase.
- Restarted?: States whether any jobs within the phase was restarted.
- Skipped: The number of jobs within the phase that was skipped.

Exceptions

Clicking the Exceptions number in the Job Timing report opens the Exception report. The Exception report is available only for jobs that have an Exceptions value greater than zero in the Job Timing report. The Exception report provides a list of exceptions encountered during the maintenance session.



The general information presented at the top are:

- Driver File: This is the name of the driver file being run when the exception occurred.
- Task Name: This is the task being performed when the exception occurred.
- Jobs Failed, then restarted successfully: This is the number of jobs that initially failed but were restarted successfully.
- Jobs Failed and skipped: This is the number of failed jobs that were skipped.

The Exception details include:

- Phase: This is the database processing phase.
- Status: This is the status of the exception.
- Product: This is the owner of the file with the exception.
- Directory: This is the location of the file.
- File: This is the file being processed when the exception occurred.

- Run Time: This is the total time the process ran.
- Restarted?: States whether the job with the exception was restarted.

AD Job Timing Report

When you run AutoPatch, AD Administration, and AutoUpgrade, they automatically generate an AD Job Timing report (adt<session_id>.lst) that shows how long it takes it takes to complete a parallel processing session, and provides information about the actions of workers as they process jobs during the session. These reports include timing statistics for the entire session, the phases in the session (AD Administration does not group jobs by phases), and individual jobs.

At any time during an AutoPatch, AD Administration, or AutoUpgrade session, you can run a script to create an AD Job Timing report that shows the progress of the current session. Or, you can go to the APPL_TOP/admin/<SID>/out directory to view an adt<session_id>.lst report from a previous session.

Note: For AutoPatch and AD Administration sessions, the adt<session_id>.lst report is very similar to the web-based Timing Report you access in Oracle Applications Manager. The information in OAM is more complete, especially for AutoPatch sessions. See OAM Timing Reports in this chapter.

AutoUpgrade Job Timing Report

We recommend that you view timing statistics for AutoPatch and AD Administration in OAM. However, this information is not available in a web-based interface for AutoUpgrade. To see results for a current AutoUpgrade session or a previous one, run the Job Timing script from the command line. The report contains the following information:

- Time-consuming jobs
- Failed jobs
- Incomplete jobs
- Total run time
- Percent usage by product
- Percent usage by phase and product

The AD Job Timing Report Interface

You run AD Job Timing Report and supply the information it needs from the command line. There are no menus or input screens.

Running AD Job Timing Report

The <output file> in these commands should not have an extension. The adtimrpt.sql script creates two files: an .lst file, which is the timing report, and a .csv file, which is currently not used. The timing report is named adt<session_id>.lst and located in $APPL_TOP/admin/<SID>/out (UNIX) or %APPL_TOP%\admin\<SID>\out (Windows).

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Run AD Job Timing report

Run the report with this command, where <session_id> is the session of the timing statistics you want to see, and <output file> is the name of the file where the statistics will be written.

UNIX:

```
$ cd $APPL_TOP/admin/<SID>/out
$ sqlplus <APPS username>/<APPS password> @$AD_TOP/admin/sql/adtimrpt.sql \
  <session id> <output file>
```

Windows:

```
C:\> cd %APPL_TOP%\admin\<SID>\out
C:\> sqlplus <APPS username>/<APPS password> @%AD_TOP%\admin\sql\adtimrpt.sql \
     <session id> <output file>
```

AD Configuration Report

The AD Configuration utility is a SQL script that reports standard information about the installed configuration of Oracle Applications. Run this task in order to debug or document the status of your installation. Running AD Configuration generates a report file (adutconf.lst) that contains the following:

- SQL*Plus PAUSE and NEWPAGE settings
- Rollback segment information
- Information about the product group
- Whether Multi-Org is installed and list of operating units
- Whether Multiple Reporting Currencies (MRC) functionality is installed
- List of registered products
- Information on all registered schemas
- Information about all installed products, including shared and dependent products
- Status of localization modules
- The base language and other installed languages
- NLS init.ora settings

The AD Configuration Report Interface

You run AD Configuration and supply the information it needs from the command line. There are no menus or input screens.

Running AD Configuration

Run this utility as follows:

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Run AD Configuration report

Use the following commands. The report output file is written to adutconf.lst in the current working directory.

UNIX:

```
$ cd $APPL_TOP/admin/<SID>/out
$ sqlplus <APPS schema username>/<APPS schema password> \
  @$AD_TOP/sql/adutconf.sql
```

Windows:

```
C:\> cd %APPL_TOP%\admin\<SID>\out
C:\> sqlplus <APPS schema username>/<APPS schema password> \
     @%AD_TOP%\sql\adutconf.sql
```

AD File Identification Report

The AD File Identification utility creates a report that identifies the version and translation level of Oracle Applications files. It is useful when collecting information about your site for Oracle Support Services.

Command Line Arguments

You may provide any number of file names as arguments on the command line. The <file *n*> arguments should be the name of any Applications text file, binary object file (extension .o for UNIX and .obj for Windows), library file (extension .a for UNIX and .lib for Windows), dynamic link library (.dll for Windows), or executable program (.exe for Windows).

When you give adident the name of a library file or executable, it lists all of the files that comprise the library or executable and their respective versions. For example:

UNIX:

```
$ adident Header $FND_TOP/lib/wfload.o $FND_TOP/lib/libfnd.a

wfload.o:
  wfload.oc            115.5.1100.3
libfnd.a:
  fdacon.lc            115.0
  fdatat.lc            115.0
  fdastr.lc            115.0
  fdaupd.lc            115.0.1100.2
  fdahmi.lc            115.0.1100.2
  fdacv.lc             115.0
  fdfutl.lc            115.4
  ....
```

Windows:

```
C:\> adident Header %FND_TOP%\lib\wfload.obj %FND_TOP%\lib\fndst.lib

wfload.obj:
  wfload.oc            115.5.1100.3
```

```
fndst.lib:
  fdacon.lc            115.0
  fdatat.lc            115.0
  fdastr.lc            115.0
  fdaupd.lc            115.0.1100.2
  fdahmi.lc            115.0.1100.2
  fdacv.lc             115.0
  fdfutl.lc            115.4
  ....
```

You can also use a "*" to identify all files in a directory (For example, *.sql to identify all SQL scripts).

The AD File Identification Report Interface

You run AD File Identification and supply the information it needs from the command line. There are no menus or input screens.

Running AD File Identification

Run this utility as follows:

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Run AD File Identification

Use the following commands. The output is written to the screen.

UNIX:

```
$ adident Header <file 1> [ <file 2> <file 3> ... ]
```

Windows:

```
C:\> adident Header <file 1> [ <file 2> <file 3> ... ]
```

AD Check Digest

The AD Check Digest utility checks the integrity of patches downloaded from Oracle. Oracle provides MD5 and SHA-1 digests for each Oracle Applications patch. The MD5 digest is a 128-bit string output that uniquely identifies the patch and the SHA-1 is a 160-bit string output. The patch digests are viewable from the Oracle*MetaLink* download page for a particular patch. Use AD Check Digest to verify whether the computed digests for the downloaded patch match the digests published on Oracle*MetaLink*.

The AD Check Digest Interface

You run AD Check Digest and supply the information it needs from the command line. There are no menus or input screens.

Parameters

The following parameters are used for running AD Check Digest.

Parameter	Definition
-file	Path and file name for the patch. This parameter is required. When the -file parameter is used without the -md5 and -sha1 parameters, AD Check Digest will compute the MD5 and SHA-1 digests for the patch.
-md5	The MD5 output from the Oracle*MetaLink* patch download page. When you use the -md5 parameter, AD Check Digest compares the MD5 value you provide with the MD5 digest computed for the patch file.
-sha1	The SHA-1 output from the Oracle*MetaLink* patch download page. When you use the -sha1 parameter, AD Check Digest compares the SHA-1 value you provide with SHA-1 digest computed for the patch file.

Running AD Check Digest

Run this utility as follows:

Step 1 Set the environment

You must set the environment in order to apply the environment variables that define your system. This task is common to many AD utilities. See Setting the Environment in Chapter 1 for the basic steps.

Step 2 Run AD Check Digest

Use the following commands. The output is written to the screen.

UNIX:

```
$ adchkdig -file <File> [ -md5 <MD5_digest> -sha1 <SHA-1_digest> ]
```

Windows:

```
C:\> adchkdig -file <File> [ -md5 <MD5_digest> -sha1 <SHA-1_digest> ]
```

Index

A

C

D

E

F

G

I

J

L

M

N

O

P

R

S

T

U

W