

Oracle® OPatch

User's Guide

10g Release 2 (10.2) for Windows and UNIX

E15294-01

June 2009

Oracle OPatch User's Guide, 10g Release 2 (10.2) for Windows and UNIX

E15294-01

Copyright © 1996, 2009, Oracle and/or its affiliates. All rights reserved.

Primary Author: Michael Zampiceni

Contributor: Phidien Nguyen

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	v
Audience	v
Documentation Accessibility	v
Conventions	vi
1 Introduction and Requirements	
OPatch Features	1-1
Requirements for OPatch	1-2
Environment Variables OPatch Uses	1-2
Backup Recommendations	1-2
2 Using Interim Patches	
Structure of Interim Patches	2-1
Interim Patch Versions	2-2
Metadata Tag	2-2
Patch Information Directory	2-2
Retrieving Interim Patches	2-2
3 OPatch Prerequisite Checks	
Prerequisite Checks in OPatch	3-1
Check ORACLE_HOME Environment Variable	3-1
Check for JRE and JDK	3-1
Check for System Space	3-1
Check for Oracle Universal Installer and OPatch Version Compatibility	3-2
Check for Patch Applicable on Operating System	3-2
Check for System Commands	3-2
Additional Prerequisite Checks for Oracle Real Application Clusters	3-2
Check for User Equivalence	3-2
Check for OPatch lsinventory	3-3
4 OPatch Utility and Commands	
OPatch Utility Basic Syntax	4-1
apply Command	4-2
napply Command	4-4

auto Command	4-7
lsinventory Command	4-8
query Command	4-12
rollback Command	4-13
version Command	4-14
5 Oracle Real Application Clusters Patching	
Specifying Node Names and Lists	5-1
All Node Patching	5-2
Rolling Patching	5-2
Minimum Downtime Patching.....	5-3
6 Resolving Patch Conflicts	
Types of Conflicts	6-1
Patch Conflict Behavior for Apply and Napply	6-4
Patch Conflict Detection and Resolution	6-5
7 Recovery and Troubleshooting	
Logging and Tracing	7-1
Restoring Oracle Homes	7-2
Recovering from a Failed Patching Session Using OPatch	7-2
Oracle Real Application Clusters (RAC) Setup	7-5
Troubleshooting OPatch	7-9
Frequently Asked Questions	7-10

Preface

OPatch is an Oracle-supplied utility that assists you with the process of applying interim patches to Oracle's software and rolling back interim patches from Oracle's software.

Audience

This guide is intended for users that have installed Oracle software products using the Oracle Universal Installer and need to subsequently apply patches. Use this guide in conjunction with any product-specific installation guides available with your Oracle product.

Documentation Accessibility

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

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

Deaf/Hard of Hearing Access to Oracle Support Services

To reach Oracle Support Services, use a telecommunications relay service (TRS) to call Oracle Support at 1.800.223.1711. An Oracle Support Services engineer will handle technical issues and provide customer support according to the Oracle service request process. Information about TRS is available at

<http://www.fcc.gov/cgb/consumerfacts/trs.html>, and a list of phone numbers is available at <http://www.fcc.gov/cgb/dro/trsphonebk.html>.

Conventions

The following text conventions are used in this document:

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

Introduction and Requirements

Patches are a small collection of files copied over an existing installation. They are associated to particular versions of Oracle Products. When applied to the correct version of an installed product, patches result in an upgraded version of the product.

The OPatch Java-based utility enables the application and rollback of interim patches to Oracle's software. OPatch is platform-independent and runs on all supported operating systems.

This chapter discusses the following introductory and advisory topics:

- [OPatch Features](#)
- [Requirements for OPatch](#)
- [Environment Variables OPatch Uses](#)
- [Backup Recommendations](#)

OPatch Features

The OPatch 10.2 utility has the following features:

- **Scalability** — OPatch is scalable to support a large number of patches.
- **Reliability** — OPatch is reliable and protects the Oracle home and inventory. It can return the Oracle home to a stable state from patch application failures. It can also easily detect patch conflicts.
- **Portability** — OPatch is compatible with all operating systems for which Oracle releases software.
- **Robust** — OPatch is very robust. It is very easy to apply a patch as well as remove it.
- **Easy to maintain** — OPatch is easy to maintain and is also extensible.
- **Support for Silent Operation** — OPatch supports silent operation. This mode enables you to run the software without any user interaction.
- **Support for Real Application Clusters** — OPatch supports Real Application Clusters and works well in that setup. It is easy to extend it to the Grid Control.
- **Easy to debug** — OPatch has various levels of logging and tracing mechanisms. It also has a debug option that helps to easily diagnose software problems.

OPatch supports the following capabilities:

- Applying an interim patch.
- Rolling back the application of an interim patch.

- Detecting conflict when applying an interim patch after previous interim patches have been applied. It also suggests the best options to resolve a conflict.
- Reporting on installed products and interim patch.

Requirements for OPatch

The OPatch utility requires the following items:

- The Oracle Universal Installer must be installed.
- The Oracle home environment variable (ORACLE_HOME) must point to a valid Oracle home directory and match the value used during installation of the Oracle home directory. See [Chapter 3, "OPatch Prerequisite Checks"](#) for more information.
- Java SDK 1.4 or higher must be available.
- Java commands for Windows must be available.
- ar, cp, fuser, and make commands for UNIX must be available.
- The library path must be set correctly for Oracle Real Application Clusters environments. OPatch uses some APIs to detect if the system is Real Application Clusters. Ensure that the library path is set correctly as follows:

```
For Solaris
LD_LIBRARY_PATH = $ORACLE_HOME/lib32:$ORACLE_HOME/lib
For HP-UX -
SHLIB_PATH=$ORACLE_HOME/lib32:/usr/lib
```

See Also: For the latest information about the OPatch utility, to check for updates, and to get latest versions, refer to My Oracle Support (formerly MetaLink) at:

<http://www.oracle.com/support/metalink/index.html>

Environment Variables OPatch Uses

OPatch uses the following environment variables:

ORACLE_HOME — Oracle home location. "[Recovering from a Failed Patching Session Using OPatch](#)"

OPATCH_DEBUG — Log level that specifies the amount of logging OPatch should perform.

OPATCH_PLATFORM_ID — Unique platform ID.

PATH — Path information.

Backup Recommendations

It is highly recommended that you back up the ORACLE_HOME before any patch operation. You can back up the ORACLE_HOME using your preferred method. You can use any method, such as zip, cp -r, tar, and cpio to compress the ORACLE_HOME.

For recovery information, see "[Recovering from a Failed Patching Session Using OPatch](#)" on page 7-2.

Using Interim Patches

Interim patches are bug fixes available to customers in response to specific bugs. They require a particular base release or patchset to be installed before they can be applied. They generally address specific bugs for a particular customer. These patches are not versioned and are generally available in a future patchset as well as the next product release.

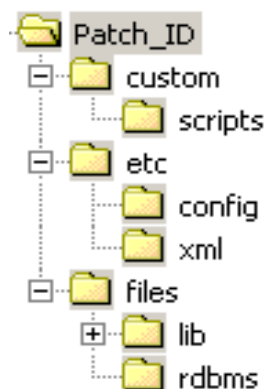
This chapter discusses the following topics on interim patches:

- [Structure of Interim Patches](#)
- [Interim Patch Versions](#)
- [Retrieving Interim Patches](#)

Structure of Interim Patches

Interim patches generally come in a zipped format. You need to unzip them before you apply a patch. [Figure 2-1](#) illustrates the structure of interim patches.

Figure 2-1 Structure of Interim Patches



Interim patches contain the following items:

- **Patch Metadata** — Information on the patch ID, bugs fixed, files affected, and actions to be performed.
- **Payload** — Files that OPatch will modify.
- **Custom Scripts** — Pre-processing and post-processing scripts that need to be run before and after patching.

Interim Patch Versions

OPatch 10.2 supports maintaining versions of patches. You can have two or more different versions of the same patch (with the same patch ID). This version information is stored in the OPatch metadata.

Metadata Tag

The metadata has a tag `date_of_patch` that stores the patch version information. The following sample shows the tag:

```
<date_of_patch year="YYYY" month="mmm" day="Day" time="Time" zone="TimeZone"/>:  
Date on which the patch was created
```

This tag records the time Oracle created the patch. If the same patch is created at a later time, this tag will record that time.

Note: This version information is the time Oracle created the patch and not the time of patch application in the host.

For example, consider a patch with ID 300200 and the `date_of_patch` tag in the patch metadata as follows:

```
<date_of_patch year="2008" month="Dec" day="23" time="04:57:13 hrs"  
zone="US/Eastern"/>
```

OPatch considers this version of the patch to be created on December 23rd, 2008 at 04:57:13 hours.

Patch Information Directory

When you apply an interim patch to an Oracle home, OPatch stores the patch information in the `$ORACLE_HOME/.patch_storage` directory. Inside this directory, separate directories are created for each patch applied to the Oracle home. You can only apply one version of the patch in the system at a given time.

You can determine the location of the patch information directory by executing the `opatch lsinventory -detail` command and looking for the patch location storage area information in the output, as shown in the following example:

```
Patch Location in Storage area:  
/home1/HOMEtoiiir571/.patch_storage/300200_Dec_23_2008_04_57_13
```

You can also find an unzipped version of the patch in the following location:

```
$ORACLE_HOME/.patch_storage/<patch_id_timestamp>/original_patch
```

Retrieving Interim Patches

Oracle releases interim patches frequently to fix a bug or a set of bugs. You can get the interim patches by specifying the patch ID in My Oracle Support (formerly MetaLink) from the following location:

```
http://www.oracle.com/support/metalink/index.html
```

OPatch Prerequisite Checks

Before you use the OPatch command and available options, you need to check that OPatch prerequisites have been fulfilled. The following sections of this chapter provide the needed information:

- [Prerequisite Checks in OPatch](#)
- [Additional Prerequisite Checks for Oracle Real Application Clusters](#)

Prerequisite Checks in OPatch

Before you invoke OPatch, perform the following prerequisite checks:

Check ORACLE_HOME Environment Variable

OPatch verifies if the Oracle home is present. You must ensure that the `ORACLE_HOME` environment variable is set to the Oracle home of the product you are trying to patch. Check your respective vendor documentation for the details to set the environment variable.

Check for JRE and JDK

OPatch requires JDK 1.4 or higher to work properly. JRE comes as a part of JDK.

OPatch 10.2 uses the `jar` utility that comes with JDK for its `jar`, `war`, and `ear` operations. OPatch searches for JDK inside the specified Oracle home. If the Oracle home does not have JDK, you have to use the `-jdk` option in OPatch to provide an alternate location. OPatch displays an error if there is a `jar/war/ear` operation and cannot locate the Java SDK location.

To find the version of JDK installed, execute the following command:

```
JDK/bin/java -version
```

Note: If the patch you are applying does not have a `jar` action, you might not need JDK, and OPatch will function acceptably with JRE alone.

Check for System Space

When OPatch processes the script for the installation of a patch, it simultaneously generates a rollback script and saves a copy of every file edited or deleted during the patching. OPatch also backs up the inventory information. Therefore, Oracle

recommends that you have sufficient system space to accommodate the patch and the backup information.

Check for Oracle Universal Installer and OPatch Version Compatibility

OPatch 10.2 requires Oracle Universal Installer 10.2 or higher to work properly. If the Oracle Universal Installer version is less than what OPatch requires, OPatch errors out.

Check for Patch Applicable on Operating System

OPatch detects if a particular patch is applicable for an operating system. If it is not applicable, it gives out a suitable error message.

Check for System Commands

OPatch supports a set of properties used for various operations of the software. You can use these properties to control the internal operations of OPatch. By default, OPatch uses the standard Java property format to specify the properties. The complete list of the default properties and their values are as follows:

```
fuser=/sbin:/usr/sbin
```

```
ar=/usr/ccs/bin/
```

```
make=/usr/bin
```

You can specify OPatch properties in the following ways:

- By using the default OPatch properties.
- By specifying the location of the user-defined properties file.
- By using the command line. The syntax is as follows:

```
PROPERTY_NAME=VALUE
```

Example: fuser=/sbin:/usr/sbin

Additional Prerequisite Checks for Oracle Real Application Clusters

For Oracle Real Application Clusters (RAC), ensure that you perform the following pre-requisite checks besides the other checks listed in the previous sections.

For specific information on Oracle RAC patching, see [Chapter 5, "Oracle Real Application Clusters Patching"](#).

Check for User Equivalence

You must ensure that the cluster systems have user equivalence set for the user installing Oracle Clusterware and Oracle Real Application Clusters. On UNIX, this means rsh , ssh or both should be set up on the cluster systems. On Windows, this means the same <domain>\<user> should have administrative privileges on all the cluster systems and the systems should be a member of the <domain>.

If the user equivalence is set properly, the following command will work properly:

```
$ rsh <nodename> date
```

Check for OPatch Lsinventory

Ensure that you are able to invoke the `opatch lsinventory -detail` command and are able to see the node information being printed out. A sample listing of the output of the command is as follows:

```
Oracle interim Patch Installer version 10.2.0.4.6
Copyright (c) 2009, Oracle Corporation. All rights reserved..
Oracle Home      : /home/racqa/102_twork/toii/toiir/toiir571/HOMEt0iir571
Central Inventory : /home/racqa/102_twork/toii/toiir/toiir571/inventory
        from      : /home/racqa/102_
twork/toii/toiir/toiir571/HOMEt0iir571/oraInst.loc
OPatch version   : 10.2.0.4.6
OUI version      : 10.2.0.4.6
OUI location     : /home/racqa/102_twork/toii/toiir/toiir571/HOMEt0iir571/oui
Log file location : /home/racqa/102_
twork/toii/toiir/toiir571/HOMEt0iir571/cfgtoollogs/opatch/opatch-2008_May_30_
01-04-52-PDT_Mon.log
Lsinventory Output file location : /home1/racqa/102_
twork/toii/toiir/toiir571/HOMEt0iir571/cfgtoollogs/opatch/lsinv/lsinventory-2008_
May_30_01-04-52-PDT_Mon.txt
-----
Installed Top-level Products (1):
Oracle Database 10g                                10.2.0.4.6
There are 1 products installed in this Oracle Home.
Installed Products (151):
Advanced Queueing (AQ) API                          10.2.0.4.6
Advanced Replication                                10.2.0.4.6
Agent Required Support Files                         10.2.0.4.6
Assistant Common Files                              10.2.0.4.6
Authentication and Encryption                       10.2.0.4.6
Bali Share                                           1.1.18.0.0
Character Set Migration Utility                     10.2.0.4.6
CSS Single-instance Common Files                     10.2.0.4.6
Data Management Services Common Files                10.2.0.4.6
Database Configuration Assistant                    10.2.0.4.6
Database SQL Scripts                                10.2.0.4.6
Database Upgrade Assistant                          10.2.0.4.6
Database Verify Utility                              10.2.0.4.6
Database Workspace Manager                          10.2.0.4.6
DBJAVA Required Support Files                       10.2.0.4.6
Documentation Required Support Files                 10.2.0.4.6
Enterprise Edition Options                          10.2.0.4.6
Enterprise Manager Agent                            10.2.0.4.6
Enterprise Manager Common Files                     10.2.0.4.6
Enterprise Manager Minimal Integration               10.2.0.4.6
Enterprise Manager plugin Common Files              10.2.0.4.6
Enterprise Manager Repository                       10.2.0.4.6
Export/Import                                       10.2.0.4.6
Extended Windowing Toolkit                          3.3.18.0.0
Generic Connectivity Common Files                    10.2.0.4.6
Generic Connectivity Using ODBC                     10.2.0.4.6
Installation Common Files                           10.2.0.4.6
Installer SDK Component                             10.2.0.1.0
iSQL*Plus                                           10.2.0.4.6
Java Naming and Directory Interface Libraries        10.2.0.4.6
Java Runtime Environment                             1.4.2.0.0
JDBC Common Files                                   10.2.0.4.6
JDBC/OCI Common Files                               10.2.0.4.6
JDBC/OCI Common Files for Instant Client            10.2.0.4.6
LDAP Required Support Files                          10.2.0.4.6
```

New Database ID	10.2.0.4.6
Object Type Translator	10.2.0.4.6
Oracle Advanced Security	10.2.0.4.6
Oracle C++ Call Interface	10.2.0.4.6
Oracle C++ Call Interface for Instant Client	10.2.0.4.6
Oracle Call Interface (OCI)	10.2.0.4.6
Oracle Client Required Support Files	10.2.0.4.6
Oracle Code Editor	1.2.1.0.0I
Oracle Containers for Java	10.2.0.4.6
Oracle Core Required Support Files	10.2.0.4.6
Oracle Data Mining	10.2.0.4.6
Oracle Database 10g	10.2.0.4.6
Oracle Database 10g	10.2.0.4.6
Oracle Database User Interface	2.2.13.0.0
Oracle Database Utilities	10.2.0.4.6
Oracle Development Kit	10.2.0.4.6
Oracle Display Fonts	9.0.2.0.0
Oracle Enterprise Manager 10g Database Control	10.2.0.4.6
Oracle Enterprise Manager Console DB	10.2.0.4.6
Oracle Extended Windowing Toolkit	3.4.28.0.0
Oracle Globalization Support	10.2.0.4.6
Oracle Help For Java	4.2.5.0.0a
Oracle Help for the Web	1.1.7.0.0a
Oracle Ice Browser	5.2.3.3.0
Oracle interMedia	10.2.0.4.6
Oracle interMedia Annotator	10.2.0.4.6
Oracle interMedia Audio	10.2.0.4.6
Oracle interMedia Client Compatibility Files	10.2.0.4.6
Oracle interMedia Client Demos	10.2.0.4.6
Oracle interMedia Client Option	10.2.0.4.6
Oracle interMedia Common Files	10.2.0.4.6
Oracle interMedia Image	10.2.0.4.6
Oracle interMedia Java Advanced Imaging	10.2.0.4.6
Oracle interMedia Java Client	10.2.0.4.6
Oracle interMedia Locator	10.2.0.4.6
Oracle interMedia Video	10.2.0.4.6
Oracle interMedia Web Client	10.2.0.4.6
Oracle Internet Directory Client	10.2.0.4.6
Oracle Internet Directory Client Common Files	10.2.0.4.6
Oracle Internet Directory Tools	10.2.0.4.6
Oracle Java Tools	10.2.0.4.6
Oracle JDBC Thin Driver for JDK 1.2	10.2.0.4.6
Oracle JDBC Thin Driver for JDK 1.4	10.2.0.4.6
Oracle JDBC Thin Driver for JDK 1.4 for Instant Client	10.2.0.4.6
Oracle JDBC/OCI Driver for JDK 1.4	10.2.0.4.6
Oracle JFC Extended Windowing Toolkit	4.2.18.0.0
Oracle JVM	10.2.0.4.6
Oracle Locale Builder	10.2.0.4.6
Oracle Message Gateway Common Files	10.2.0.4.6
Oracle Net	10.2.0.4.6
Oracle Net Configuration Assistant	10.2.0.4.6
Oracle Net Listener	10.2.0.4.6
Oracle Net Manager	10.2.0.4.6
Oracle Net Required Support Files	10.2.0.4.6
Oracle Net Services	10.2.0.4.6
Oracle Notification Service	9.0.4.0.0
Oracle OLAP	10.2.0.4.6
Oracle OLAP API	10.2.0.4.6
Oracle OLAP Catalog	10.2.0.4.6
Oracle One-Off Patch Installer	10.2.0.1.0

Oracle Partitioning	10.2.0.4.6
Oracle Programmer	10.2.0.4.6
Oracle RAC Required Support Files	10.2.0.4.6
Oracle RAC Required Support Files-HAS	10.2.0.4.6
Oracle Real Application Clusters	10.2.0.4.6
Oracle Spatial	10.2.0.4.6
Oracle Starter Database	10.2.0.4.6
Oracle Text	10.2.0.4.6
Oracle UIX	2.1.21.0.0a
Oracle Ultra Search Common Files	10.2.0.4.6
Oracle Ultra Search Middle-Tier	10.2.0.4.6
Oracle Ultra Search Server	10.2.0.4.6
Oracle Universal Installer	10.2.0.1.0
Oracle Wallet Manager	10.2.0.4.6
Oracle XML Developer's Kit	10.2.0.4.6
Oracle XML Runtime Components	10.2.0.4.6
Oracle XML SQL Utility	10.2.0.4.6
Oracle10g Real Application Clusters Common Files	10.2.0.4.6
Parser Generator Required Support Files	10.2.0.4.6
Perl Interpreter	5.6.1.0.2d
PL/SQL	10.2.0.4.6
PL/SQL Embedded Gateway	10.2.0.4.6
PL/SQL Required Support Files	10.2.0.4.6
Platform Required Support Files	10.2.0.4.6
Precompiler Common Files	10.2.0.4.6
Precompiler Required Support Files	10.2.0.4.6
Pro*C/C++	10.2.0.4.6
RDBMS Required Support Files	10.2.0.4.6
Recovery Manager	10.2.0.4.6
regexp	2.1.9.0.0
Required Support Files	10.2.0.4.6
Sample Schema	10.2.0.4.6
Secure Socket Layer	10.2.0.4.6
Secure Socket Layer	10.2.0.4.6
SQL*Loader	10.2.0.4.6
SQL*Plus	10.2.0.4.6
SQL*Plus Required Support Files	10.2.0.4.6
SQLJ Runtime	10.2.0.4.6
SSL Required Support Files	10.2.0.4.6
SSL Required Support Files for InstantClient	10.2.0.4.6
Sun JDK	1.4.2.0.0
Sun JDK extensions	9.0.4.0.0
Utilities Common Files	10.2.0.4.6
Visigenics ORB	3.4.0.0.0
XDK Required Support Files	10.2.0.4.6
XML	10.2.0.4.6
XML Class Generator for C++	10.2.0.4.6
XML Class Generator for Java	10.2.0.4.6
XML Parser for C	10.2.0.4.6
XML Parser for C++	10.2.0.4.6
XML Parser for Java	10.2.0.4.6
XML Parser for Oracle JVM	10.2.0.4.6
XML Parser for PL/SQL	10.2.0.4.6
XML Transviewer Beans	10.2.0.4.6
XML Transx	10.2.0.4.6
XSQL Servlet	10.2.0.4.6

There are 151 products installed in this Oracle Home.

Intermin patches (1) :

Patch 3811942 : applied on Mon May 30 00:59:33 PDT 2008

Created on 31 Aug 2007, 12:06:28 hrs US/Pacific

```

Bugs fixed:
  3811942
Files Touched:
  /hosp.o --> ORACLE_HOME/lib/libagent10.a
  /pesblt.o --> ORACLE_HOME/lib/libpls10.a
  /kgl.o --> ORACLE_HOME/lib/libgeneric10.a
  /qcpi6.o --> ORACLE_HOME/lib/libgeneric10.a
  ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/iextjob
  ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/iextjobo
  ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/client_sharedlib
  /hosp.o --> ORACLE_HOME/lib/libserver10.a
  /prse.o --> ORACLE_HOME/lib/libserver10.a
  /prsa.o --> ORACLE_HOME/lib/libserver10.a
  /prsf.o --> ORACLE_HOME/lib/libserver10.a
  /prssz.o --> ORACLE_HOME/lib/libserver10.a
  /kprc.o --> ORACLE_HOME/lib/libserver10.a
  /qmhdr.o --> ORACLE_HOME/lib/libserver10.a
  /pesblt.o --> ORACLE_HOME/lib/libpls10.a
  /qcpi6.o --> ORACLE_HOME/lib/libgeneric10.a
  prvtpexp.plb --> ORACLE_HOME/rdbms/admin/prvtpexp.plb
  sjsex.o --> ORACLE_HOME/rdbms/lib/sjsex.o
  ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/iextjob
  ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/iextjobo
  ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/ioracle
  ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/client_sharedlib
  driload.pkh --> ORACLE_HOME/ctx/admin/driload.pkh
  /oracle/sysman/assistants/util/em/EMConfigStep.class --> ORACLE_
HOME/assistants/jlib/assistantsCommon.jar
  /oracle/sysman/assistants/util/sqlEngine/SQLEngine.class --> ORACLE_
HOME/assistants/jlib/assistantsCommon.jar
  /oracle/sysman/assistants/dbca/backend/DBEntryStep.class --> ORACLE_
HOME/assistants/dbca/jlib/dbca.jar
  /oracle/sysman/assistants/dbca/backend/EMConfigStep.class --> ORACLE_
HOME/assistants/dbca/jlib/dbca.jar
  /oracle/sysman/assistants/dbca/backend/PostDBCreationStep.class --> ORACLE_
HOME/assistants/dbca/jlib/dbca.jar
  /oracle/sysman/emcp/EMConfig.class --> ORACLE_HOME/jlib/emca.jar
  /oracle/sysman/emcp/IEMCAConstants.class --> ORACLE_HOME/jlib/emca.jar
  /oracle/sysman/emcp/EMConfig.class --> ORACLE_HOME/sysman/jlib/emCORE.jar
  /oracle/sysman/emcp/IEMCAConstants.class --> ORACLE_
HOME/sysman/jlib/emCORE.jar
  /oracle/sysman/emSDK/conf/ConfigManager.class --> ORACLE_
HOME/sysman/jlib/emCORE.jar
  /oracle/sysman/emSDK/eml/EmlConstants.class --> ORACLE_
HOME/sysman/jlib/emCORE.jar
  /oracle/sysman/util/pref/PrefUtil.class --> ORACLE_
HOME/sysman/jlib/emCORE.jar
  emctl.pl --> ORACLE_HOME/emdw/bin/emctl.pl
  /ncrfipm.o --> ORACLE_HOME/lib/libnro10.a
  libsqlplus.a --> ORACLE_HOME/lib/libsqlplus.a
  libsqlplus.so --> ORACLE_HOME/lib/libsqlplus.so
  libisqlplus.a --> ORACLE_HOME/lib/libisqlplus.a
  libisqlplus.so --> ORACLE_HOME/lib/libisqlplus.so
  mdprivs.sql --> ORACLE_HOME/md/admin/mdprivs.sql
  prvtmpd.plb --> ORACLE_HOME/md/admin/prvtmpd.plb
  sdolrsmd.sql --> ORACLE_HOME/md/admin/sdolrsmd.sql
  prvtcchk.plb --> ORACLE_HOME/md/admin/prvtcchk.plb
  wk0acl.pkh --> ORACLE_HOME/ultrasearch/admin/wk0acl.pkh
  wk0acl.plb --> ORACLE_HOME/ultrasearch/admin/wk0acl.plb
  wk0adm.pkh --> ORACLE_HOME/ultrasearch/admin/wk0adm.pkh

```



```
wk0adm.plb --> ORACLE_HOME/ultrasearch/admin/wk0adm.plb
wk0snapshot.plb --> ORACLE_HOME/ultrasearch/admin/wk0snapshot.plb
wk0util.pkh --> ORACLE_HOME/ultrasearch/admin/wk0util.pkh
wk0util.plb --> ORACLE_HOME/ultrasearch/admin/wk0util.plb
Patch Location in Inventory:
  /home1/racqa/102_
twork/toii/toiir/toiir571/HOMEttoiir571/inventory/oneoffs/3811942
Patch Location in Storage area:
  /home1/racqa/102_twork/toii/toiir/toiir571/HOMEttoiir571/.patch_
storage/3811942_Aug_31_2007_12_06_28
Rac system comprising of multiple nodes
  Local node = stadu56
  Remote node = stacg16
-----
OPatch succeeded.
```

If you do not find the node information correctly printed out, you need to update the node list. For more information on updating the node list, see the *Oracle Universal Installer User's Guide*.

OPatch Utility and Commands

The OPatch utility is a tool that allows the application and rollback of interim patches to Oracle products. This chapter provides information on using OPatch to apply patches.

This chapter includes the following topics:

- [OPatch Utility Basic Syntax](#)
- [apply Command](#)
- [napply Command](#)
- [auto Command](#)
- [lsinventory Command](#)
- [query Command](#)
- [rollback Command](#)
- [version Command](#)

OPatch Utility Basic Syntax

The OPatch utility is located in the `<Path_to_Oracle_Home>/OPatch` directory. You can run it with various commands and options. The following command shows the syntax for the OPatch utility:

```
<Path_to_OPatch>/opatch <command> [-options]
```

In the preceding command, the following variables are used:

- *command*: The OPatch command, described in [Table 4-1](#):
- *options*: The command-line arguments for the command, which are described in the following sections.

Table 4-1 *OPatch Commands*

Command	Description
apply	Installs an interim patch. Refer to " apply Command " for more information.
napply	Installs <i>n</i> number of patches (hence napply). Refer to " napply Command " for more information.
auto	Applies Oracle Clusterware patches. Refer to " auto Command " for more information.

Table 4–1 (Cont.) OPatch Commands

Command	Description
lsinventory	Lists what is currently installed on the system. Refer to " lsinventory Command " for more information.
query	Queries a given patch for specific details. Refer to " query Command " for more information.
rollback	Removes an interim patch. Refer to " rollback Command " for more information.
version	Prints the current version of the patch tool. Refer to " version Command " for more information.

To view additional information for any command, use the following command:

```
<Path_to_OPatch>/opatch command -help
```

If using Perl, use the following command:

```
perl opatch.pl command -help
```

To show the full syntax of the -help option, enter `opatch -h` to view the following display:

```
Usage: opatch [ -help ] [ -r[eport] ] [ command ]
```

```

command := auto
          apply
          lsinventory
          napply
          nrollback
          rollback
          query
          version
          prereq
          util
    
```

```

<global_arguments> := -help      Displays the help message for the command.
                    -report     Print the actions without executing.
    
```

example:

```

'opatch -help'
'opatch auto -help'
'opatch apply -help'
'opatch lsinventory -help'
'opatch napply -help'
'opatch nrollback -help'
'opatch rollback -help'
'opatch prereq -help'
'opatch util -help'
    
```

apply Command

The `apply` command applies an interim patch to a specified Oracle home. The `ORACLE_HOME` environment variable must be set to the Oracle home to be patched. Use following syntax for this command:

```

<Path_to_OPatch>/opatch apply [-delay (value)] [-force] \
[-invPtrLoc (path)] [-jdk (location)] [-jre (location)] [-local] \
[-minimize_downtime] [-no_bug_superset] [-no_inventory] \
    
```

```
[-oh (Oracle Home location)] \
[-post (options to be passed into post) [-opatch_post_end]] \
[-pre (options to be passed into pre) [-opatch_pre_end]] \
[-retry (value)] [-silent] [-verbose] [-no_relink] \ [-no_sysmod (patch
location)] [-remote_nodes (comma separated node names)] [-local_node (node
_name)] [patch_location]
```

Table 4–2 describes the options available for the `apply` command.

Table 4–2 *apply Command Options*

Option	Description
<code>delay</code>	Specifies how many seconds to wait before attempting to lock the inventory in the case of a previous failure. You can use this option only if you specify the <code>-retry</code> option.
<code>force</code>	Removes conflicting patches from the system. If a conflict exists that prevents the patch from being applied, you can use the <code>-force</code> option to apply the patch.
<code>invPtrLoc</code>	Specifies the location of the <code>oraInst.loc</code> file. This option is needed when the <code>-invPtrLoc</code> argument was used during installation. Oracle recommends using the default Central Inventory for a platform.
<code>jdk</code>	Specifies the location of a particular JDK (jar) to use instead of the default location under the Oracle home directory. You cannot use the <code>-jdk</code> and <code>-jre</code> options together.
<code>jre</code>	Specifies the location of a particular JRE (Java) to use instead of the default location under the Oracle home directory. You cannot use the <code>-jdk</code> and <code>-jre</code> options together.
<code>local</code>	Specifies that the OPatch utility should patch the local node and update the inventory of the local node. It does not propagate the patch or inventory update to other nodes. You can use this option on Oracle Real Application Clusters environments and non-clustered environments. If an entire cluster is shut down before patching, you can use this argument for non-rolling patches.
<code>local_node</code>	Specifies the local node for this cluster to the OPatch utility. You can use this option on Oracle Real Application Clusters environments.
<code>minimize_downtime</code>	Specifies the order of nodes to be patched by the OPatch utility. This option only applies to Oracle Real Application Clusters environments. You cannot use it with the <code>-local</code> option or a rolling patch.
<code>no_bug_superset</code>	Specifies to error out if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory.
<code>no_inventory</code>	Bypasses the inventory for reading and updates. You cannot use this option with the <code>-local</code> option. This option puts the installation into an unsupported state.
<code>no_sysmod</code>	Specifies that the OPatch utility need not update the files in the system. It will only update the inventory.
<code>no_relink</code>	This option does not perform any make operation. You can use it during multiple patch applications and to perform the linking step only once. OPatch does not keep track of the make operations it did not perform. You need to make sure to execute OPatch without this option at the end for compilation.

Table 4–2 (Cont.) apply Command Options

Option	Description
oh	Specifies the Oracle home directory to use instead of the default.
opatch_post_end	Marks the end of the <code>post</code> option. This option is used with the <code>post</code> option. If this argument is not used, everything after <code>post</code> is passed into <code>post</code> .
opatch_pre_end	Marks the end of the <code>pre</code> options. This option is used with the <code>pre</code> option. If you do not use this argument, everything after <code>pre</code> is passed into <code>pre</code> .
post	Specifies the parameters to be passed inside the <code>post</code> script besides the standard parameters.
pre	Specifies the parameters to be passed inside the <code>pre</code> script besides the standard parameters.
remote_nodes	Specifies the list of remote nodes to the OPatch utility. You can use this option on Oracle Real Application Clusters environments.
report	Prints the action to the screen without executing it.
retry	Specifies how many times the OPatch utility should try when there is an inventory lock failure.
silent	Suppresses user interaction, and defaults any answers to "yes."
verbose	Prints output to the screen as well as to the log file.

Note: If a patch consists of SQL changes, follow the instructions in the patch readme, included with the patch to apply the SQL scripts.

napply Command

This command applies interim patches to several Oracle homes at the same time.

Syntax

Use the following syntax for this command:

```
opatch napply [patch_location] [-id comma-separated list of patch IDs]
    [-delay <value> ] [ -force ]
    [-invPtrLoc <Path to oraInst.loc> ]
    [-jdk <LOC> ] [-jre <LOC> ] [ -local ]
    [-minimize_downtime ] [-no_bug_superset ]
    [-no_inventory ] [-oh <ORACLE_HOME> ]
    [-retry <value> ] [-silent ]
    [-verbose ] [-no_relink]
    [-pre <parameters for the pre script in
    escaped double quotes> [-opatch_pre_end] ]
    [-post <parameters for the post script in
    escaped double quotes> [-opatch_post_end] ]
    [-no_sysmod]
    [ -property_file <Path to property file> ]
    [ -local_node <Local node name> ]
    [ -remote_nodes <List of remote nodes (node1,node2)> ]
    [ -all_nodes ]
    [ -phBaseFile <Path to the file containing the
    location of the patches to be applied> ]
    [-skip_subset] [-skip_duplicate] [-report]
```

Examples

- The following example applies all patches under the `<patch_location>` directory:

```
opatch napply <patch_location>
```

- The following example applies patches 1, 2, and 3 that are under the `<patch_location>` directory:

```
opatch napply <patch_location> -id 1,2,3
```

- The following example applies all patches under the `<patch_location>` directory. OPatch skips duplicate patches and subset patches (patches under `<patch_location>` that are subsets of patches installed in the Oracle home).

```
opatch napply <patch_location> -skip_subset -skip_duplicate
```

See the description for the `skip_subset` option in [Table 4–3](#) for more information.

- The following example applies patches 1, 2, and 3 that are under the `<patch_location>` directory. OPatch skips duplicate patches and subset patches (patches under `<patch_location>` that are subsets of patches installed in the Oracle home).

```
opatch napply <patch_location> -id 1,2,3 -skip_subset -skip_duplicate
```

See the description for the `skip_subset` option in [Table 4–3](#) for more information.

Options

[Table 4–3](#) lists the options available for this command.

Table 4–3 Napply Command Options

Option	Description
<code>all_nodes</code>	Applies the patch using the all-node mode.
<code>delay</code>	Specifies how many seconds to wait before attempting to lock the inventory again for a previous failure. You can use this option only if you specify the <code>retry</code> option.
<code>force</code>	Removes conflicting patches from the system. If a conflict exists that prevents the patch from being applied, you can use this option to apply the patch. OPatch removes all the conflicting patches before applying the current patch.
<code>invPtrLoc</code>	Specifies the location of the <code>oraInst.loc</code> file. The <code>invPtrLoc</code> option is needed when this option is used during installation. Oracle recommends the use of the default Central Inventory for a platform.
<code>jdk</code>	Instructs OPatch to use JDK (jar) from the specified location instead of the default location under the Oracle home directory. If you do not specify the <code>jre</code> option, JVM is executed from the <code>jdk</code> location.
<code>jre</code>	Instructs OPatch to use JRE (Java) from the specified location instead of the default location under the Oracle home directory. You cannot specify the <code>jdk</code> and <code>jre</code> options together.

Table 4–3 (Cont.) Napply Command Options

Option	Description
local	Specifies that OPatch should patch the local node and update the inventory of the local node. It does not propagate the patch or inventory update to other nodes. You can use this option on Oracle Real Application Clusters environments and non-clustered environments. If an entire cluster is shut down before patching, you can use this option for non-rolling patches.
local_node	Tells OPatch the local node for this cluster. You can use this option on Oracle Real Application Clusters environments.
minimize_downtime	Specifies the order of nodes that OPatch should patch. This option only applies to Oracle Real Application Clusters environments. You cannot use it with the <code>-local</code> option with a rolling patch.
no_bug_superset	Specifies to error out if the current patch's bugs-to-fix is a superset (or same set) of an installed patch's bugs-fixed in the Oracle home directory.
no_inventory	Bypasses the inventory for reading and updates. You cannot use this option with the <code>local</code> option. This option places the installation into an unsupported state.
no_relink	This option does not perform any make operations. You can use it during multiple patch applications and to perform the linking step only once. OPatch does not keep track of the make operations it did not perform. You need to make sure to execute OPatch without this option at the end for compilation.
no_sysmod	Specifies that OPatch does not need to update the files in the system. It only updates the inventory. It also does not execute the pre and post scripts.
oh	Specifies the Oracle home directory to use instead of the default. This takes precedence over the environment variable <code>ORACLE_HOME</code> .
opatch_post_end	Marks the end of the <code>post</code> option. You use this option with the <code>post</code> option. If you do not use this option, everything after <code>post</code> until the end of the command is passed into <code>post</code> .
opatch_pre_end	Marks the end of the <code>pre</code> options. You use this option with the <code>pre</code> option. If you do not use this option, everything after <code>pre</code> until the end of the command is passed into <code>pre</code> .
<i>Patch Location</i>	Indicates the path to the patch location. If you do not specify the location, OPatch assumes the current directory is the patch location.
phBaseFile	If you do not specify <code><patch_location></code> , use this option to point OPatch to a file containing a list of patches to be n-applied. Each line in the file points to a location of a patch.
post	Specifies the parameters to be passed to the <code>post</code> script. This script is executed after the patch is applied. You need to enclose the values for this option in double-quotes.
pre	Specifies the parameters to be passed to the <code>pre</code> script. This script is executed before the patch is applied. You need to enclose the values for this option in double-quotes.
property_file	Specifies the user-defined property file for OPatch to use. The path to the property file should be absolute. This property file takes precedence over the one that OPatch supplies.

Table 4–3 (Cont.) Napply Command Options

Option	Description
remote_nodes	Tells OPatch the list of remote nodes. You can use this option on Oracle Real Application Clusters environments. The node names must be separated with commas, but without spaces.
report	Prints the action to the screen without executing it.
retry	Tells OPatch how many times it should retry when there is an inventory lock failure.
silent	Suppresses user interaction, and defaults any answers to "yes."
skip_duplicate	Skips patches to be applied that are duplicates of other patches installed in the Oracle home. Two patches are duplicates if they fix the same set of bugs.
skip_subset	Skips patches to be applied that are subsets of other patches installed in the Oracle home. One patch is a subset of another patch if the former fixes a subset of bugs fixed by the latter. For example, if you used <code>napply</code> yesterday for patch A that fixed bugs 1 and 2, then you use <code>napply</code> today with this option for patch B that fixes bug 1 and patch C that fixes bugs 1, 2, and 3, then subset patch A is skipped, and patch C then becomes a superset of patch A.
verbose	Prints additional OPatch output to the screen as well as to the log file.

auto Command

Ordinarily, a Clusterware patch requires several manual steps before and after you apply the patch, such as:

- Stopping all dependent databases
- Stopping Clusterware resources
- Running pre-patch scripts
- Shutting down Clusterware
- Running post-patch scripts
- Starting Clusterware and dependent databases

The `opatch auto` command automates all of these tasks for patching the CRS home and all other applicable RDBMS homes.

Syntax

Use the following syntax for this command:

```
<path_to_opatch>/opatch auto
    [-rollback [patch_location]]
    [[patch_location]-oh <path_to_oracle_home1>,<path_to_oracle_home2>...] |
    [[patch_location]-och <path_to_crs_home>]
```

... where *patch_location* is path to the location for the patch. If you do not specify the patch location, the current directory is considered the patch location.

Options

[Table 4–4](#) lists the options available for this command.

Table 4–4 Auto Command Options

Option	Description
rollback	Rolls back the patch rather than applying it.
oh	Comma-separated Oracle homes to patch. The default is all applicable Oracle homes. Use this option to patch RDBMS homes where no database is registered.
och	Path of the Oracle Clusterware home. Use this option to patch only Oracle Clusterware homes where Oracle Clusterware has been stopped already. Do not use this option for Oracle Clusterware with a CRS stack that is up.

Examples

- The following example applies a patch with an unzipped patch location to all applicable Oracle homes on the system:

```
opatch auto <patch_location>
```
- The following example rolls back the patch from all the applicable Oracle homes on the system:

```
opatch auto -rollback <patch_location>
```
- The following example patches a selective list of Oracle homes:

```
opatch auto <patch_location> -oh /tmp/oh1,/tmp/oh2,/tmp/oh3
```
- The following example only patches the CRS home when the Oracle Clusterware stack is down.

```
opatch auto <patch_location> -och /tmp/ora_crs_home
```

lsinventory Command

The `lsinventory` command reports what has been installed on the system for a particular Oracle home directory, or for all installations. The following syntax is used for this command:

```
<Path_to_OPatch>/opatch lsinventory [-all] [-bugs_fixed asc|desc] [-delay (value)]
[-detail] [-invPrtLoc (path)] \
[-jre (location)] [-patch asc|desc] [-oh (Oracle Home location)] [-retry (value)]
```

Table 4–5 describes the options available for the `lsinventory` command.

Table 4–5 lsinventory Command Options

Option	Description
all	Reports the name and installation directory for each Oracle home directory found.

Table 4–5 (Cont.) Isinventory Command Options

Option	Description
bugs_fixed	Reports bugs fixed by installed patches in a tabular format. Besides the bugs fixed, the report also displays the installed patches, installed times, and bug descriptions. The fixed bugs are sorted per installed patch. The default display is patches in descending order based on installed time and ascending order of bugs within each patch. You can use 'asc' or 'desc' with this option to enforce sort order on bugs within each patch. You can use this option with the patch or patch_id option to obtain sort orders with installed patches.
delay	Specifies how many seconds to wait before attempting to lock the inventory in the case of a previous failure. You can use this option only if the -retry option is specified.
detail	Reports the installed products and other details. You cannot use this option with the -all option.
invPtrLoc	Specifies the location of the oraInst.loc file. This option is needed when the invPtrLoc option was used during installation. Oracle recommends using the default Central Inventory for a platform.
jre	Specifies the location of a particular JRE (Java) to use instead of the default location under the Oracle home directory.
oh	Specifies the Oracle home directory to use instead of the default directory.
patch	Lists the patch IDs installed in the Oracle home in ascending (asc) or descending (desc) order, which is the default, based on installed time.
retry	Specifies how many times the OPatch utility should try when there is an inventory lock failure.

-detail Option Example

The following example shows the output of `opatch lsinventory -detail`:

```
Oracle interim Patch Installer version 10.2.0.4.6
Copyright (c) 2009, Oracle Corporation. All rights reserved..
Oracle Home      : /home/oracle_TEST/product/10.2.0/db_1
Central Inventory : /home/OUIHome_Opatch
    from         : /home/oracle_TEST/product/10.2.0/db_1/oraInst.loc
OPatch version   : 10.2.0.4.6
OUI version      : 10.2.0.4.6
OUI location     : /home/oracle_TEST/product/10.2.0/db_1/oui
Log file location : /home/oracle_
TEST/product/10.2.0/db1/cfgtoollogs/opatch/opatch-2008_May_25_11-09-34-IST_Wed.log
Patch history file: /scratch/userid/newDB/cfgtoollogs/opatch/opatch_history.txt
Lsinventory Output file location : /home/oracle_TEST/product/10.2.0/db_
1/cfgtoollogs/opatch/lsinv/lsinventory-2008_May_25_11-09-34-IST_Wed.txt
-----
Installed Top-level Products (1):
Oracle Database 10g                      10.2.0.4.6
There are 1 products installed in this Oracle Home.
Installed Products (10):
Agent Required Support Files             10.2.0.4.6
Assistant Common Files                   10.2.0.4.6
Bali Share                               1.1.18.0.0
Buildtools Common Files                  10.2.0.4.6
Character Set Migration Utility           10.2.0.4.6
```

```

Database Configuration and Upgrade Assistants          10.2.0.4.6
Database SQL Scripts                                  10.2.0.4.6
Database Workspace Manager                            10.2.0.4.6
DBJAVA Required Support Files                         10.2.0.4.6
Enterprise Edition Options                            10.2.0.4.6

```

There are 10 products installed in this Oracle Home.

Interim patches (1) :

Patch 111000 : applied on Mon May 23 19:44:08 IST 2008

Created on 27 Jul 2007, 05:43:46 hrs PST8PDT

Bugs fixed: 111000

Files Touched:

```

/qmtest.o --> ORACLE_HOME/lib/libserver11.a
libmapsym.so --> ORACLE_HOME/lib/libmapsym.so
ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/ioracle
/oracle/xml/jaxb/orajaxb.class --> ORACLE_HOME/lib/xml.jar

```

Patch Location in Inventory:

/home/oracle_TEST/product/10.2.0/db_1/inventory/oneoffs/111000

Patch Location in Storage area:

/home/oracle_TEST/product/10.2.0/db_1/.patch_storage/111000_Jul_27_2007_05_43_46

OPatch succeeded.

-bugs_fixed Option Example

The following example shows the output of `opatch lsinventory`

`-bugs_fixed asc`:

```

Oracle interim Patch Installer version 10.2.0.4.6
Copyright (c) 2009, Oracle Corporation. All rights reserved..
Oracle Home      : /home/oracle_TEST/product/10.2.0/db_1
Central Inventory : /home/OUIHome_Opatch
                  from   : /home/oracle_TEST/product/10.2.0/db_1/oraInst.loc
OPatch version   : 10.2.0.4.6
OUI version      : 10.2.0.4.6
OUI location     : /home/oracle_TEST/product/10.2.0/db_1/oui
Log file location : /home/oracle_
TEST/product/10.2.0/db1/cfgtoollogs/opatch/opatch-2008_May_25_11-09-34-IST_Wed.log
Patch history file: /scratch/userid/newDB/cfgtoollogs/opatch/opatch_history.txt
Lsinventory Output file location : /home/oracle_TEST/product/10.2.0/db_
1/cfgtoollogs/opatch/lsinv/lsinventory-2008_May_25_11-09-34-IST_Wed.txt
-----

```

Installed Top-level Products (2) :

```

Oracle Database 10g          10.2.0.4.6
Oracle Database 10g Release 2 Patch Set 2  10.2.0.4.6

```

There are 2 products installed in this Oracle Home.

List of Bugs fixed by Installed Patches:

Bug	Fixed by Patch	Installed at	Description
---	-----	-----	-----
1000000	6079591	Mon Oct 13 02:03:42 PDT 2008	test bug
6079591	6079591	Mon Oct 13 02:03:42 PDT 2008	MLR BUG FOR 10.2.0:.3 FOR CPU:JUL2:007
300500	300500	Fri Sep 05 02:25:34 PDT 2008	Demo bug for patching files
300501	300500	Fri Sep 05 02:25:34 PDT 2008	Demo bug for patching files
300502	300500	Fri Sep 05 02:25:34 PDT 2008	Demo bug for patching files
6121268	6121268	Tue Aug 19 23:32:33 PDT 2008	

```

DB-10.2.0.3-MOLECULE-007-CPUJUL2007
6121266      6121266      Tue Aug 19 23:32:27 PDT 2008
DB-10.2.0.3-MOLECULE-018-CPUJUL2007
6121264      6121264      Tue Aug 19 23:32:22 PDT 2008
DB-10.2.0.3-MOLECULE-017-CPUJUL2007
6121263      6121263      Tue Aug 19 23:32:14 PDT 2008
DB-10.2.0.3-MOLECULE-016-CPUJUL2007
.....
.....
(Middle section of report is intentionally excluded.)
.....
.....
6121248      6650096      Tue Feb 12 05:50:48 PST 2008
DB-10.2.0.3-MOLECULE-015-CPUJUL2007
6650096      6650096      Tue Feb 12 05:50:48 PST 2008
DB-10.2.0.3-MOLECULE-036-CPUJAN2008
6121247      6650095      Tue Feb 12 05:50:41 PST 2008
DB-10.2.0.3-MOLECULE-006-CPUAPR2007
6397946      6650095      Tue Feb 12 05:50:41 PST 2008
DB-10.2.0.3-MOLECULE-031-CPUOCT2007
6650095      6650095      Tue Feb 12 05:50:41 PST 2008
DB-10.2.0.3-MOLECULE-035-CPUJAN2008
6650081      6650081      Tue Feb 12 05:50:35 PST 2008
DB-10.2.0.3-MOLECULE-034-CPUJAN2008
6646853      6646853      Tue Feb 12 05:50:28 PST 2008      MLR BUG FOR 10.2.0.3 FOR
CPUJAN2008
6452863      6452863      Tue Feb 12 05:50:12 PST 2008      TRACKING BUG FOR CPUJUL2007
-----
OPatch succeeded.

```

-patch desc Option Example

The following example shows the output of `opatch lsinventory -patch desc`:

```

Oracle interim Patch Installer version 10.2.0.4.6
Copyright (c) 2009, Oracle Corporation. All rights reserved..
Oracle Home      : /home/oracle_TEST/product/10.2.0/db_1
Central Inventory : /home/OUIHome_Opatch
      from       : /home/oracle_TEST/product/10.2.0/db_1/oraInst.loc
OPatch version   : 10.2.0.4.6
OUI version      : 10.2.0.4.6
OUI location     : /home/oracle_TEST/product/10.2.0/db_1/oui
Log file location : /home/oracle_
TEST/product/10.2.0/db1/cfgtoollogs/opatch/opatch-2008_May_25_11-09-34-IST_Wed.log
Patch history file: /scratch/userid/newDB/cfgtoollogs/opatch/opatch_history.txt
Lsinventory Output file location : /home/oracle_TEST/product/10.2.0/db_
1/cfgtoollogs/opatch/lsinv/lsinventory-2008_May_25_11-09-34-IST_Wed.txt
-----
Interim patches (39) :
Patch 6079591      : applied on Mon Oct 13 02:03:42 PDT 2008
      Created on 21 Jun 2007, 03:42:18 hrs PST8PDT
      Bugs fixed:
          6079591, 1000000

Patch 300500      : applied on Fri Sep 05 02:25:34 PDT 2008
      Created on 07 Nov 2005, 04:57:14 hrs US/Eastern
      Bugs fixed:
          300500, 300501, 300502
-----
OPatch succeeded.

```

-detail Option Example

The following example shows the output of `opatch lsinventory -detail`:

```
Oracle interim Patch Installer version 10.2.0.4.6
Copyright (c) 2009, Oracle Corporation. All rights reserved..
Oracle Home      : /home/oracle_TEST/product/10.2.0/db_1
Central Inventory : /home/OUIHome_Opatch
    from          : /home/oracle_TEST/product/10.2.0/db_1/oraInst.loc
OPatch version   : 10.2.0.4.6
OUI version      : 10.2.0.4.6
OUI location     : /home/oracle_TEST/product/10.2.0/db_1/oui
Log file location : /home/oracle_
TEST/product/10.2.0/db1/cfgtoollogs/opatch/opatch-2008_May_25_11-09-34-IST_Wed.log
Lsinventory Output file location : /home/oracle_TEST/product/10.2.0/db_
1/cfgtoollogs/opatch/lsinv/lsinventory-2008_May_25_11-09-34-IST_Wed.txt
-----
Installed Top-level Products (1):
Oracle Database 10g                               10.2.0.4.6
There are 1 products installed in this Oracle Home.
Installed Products (10):
Agent Required Support Files                       10.2.0.4.6
Assistant Common Files                             10.2.0.4.6
Bali Share                                         1.1.18.0.0
Buildtools Common Files                           10.2.0.4.6
Character Set Migration Utility                   10.2.0.4.6
Database Configuration and Upgrade Assistants     10.2.0.4.6
Database SQL Scripts                              10.2.0.4.6
Database Workspace Manager                        10.2.0.4.6
DBJAVA Required Support Files                     10.2.0.4.6
Enterprise Edition Options                        10.2.0.4.6
There are 10 products installed in this Oracle Home.
Interim patches (1) :
Patch 102000      : applied on Mon May 23 19:44:08 IST 2008
Created on 27 Jul 2007, 05:43:46 hrs PST8PDT
Bugs fixed: 102000
Files Touched:
/qntest.o --> ORACLE_HOME/lib/libserver10.a
libmapsym.so --> ORACLE_HOME/lib/libmapsym.so
ins_rdbms.mk --> ORACLE_HOME/rdbms/lib/ioracle
/oracle/xml/jaxb/orajaxb.class --> ORACLE_HOME/lib/xml.jar
Patch Location in Inventory:
/home/oracle_TEST/product/10.2.0/db_1/inventory/oneoffs/102000
Patch Location in Storage area:
/home/oracle_TEST/product/10.2.0/db_1/.patch_storage/102000_Jul_27_2007_05_43_46
-----
OPatch succeeded.
```

query Command

The query command queries a specific patch for specific details. It provides information about the patch and the system being patched. The following syntax is used for this command:

```
<Path_to_OPatch>/opatch query [-all] [-jre (Location)] [-jdk (Location)]\
[-oh (Location)] [patch_location]
```

[Table 4-6](#) lists the options available for the query command.

Table 4–6 query Command Options

Option	Description
all	Retrieves all information about a patch. This is equivalent to setting all options.
jdk	Specifies the location of a particular JDK (jar) to use instead of the default location under the Oracle home directory. You cannot use the -jdk and -jre options together.
jre	Specifies the location of a particular JRE (Java) to use instead of the default location under the Oracle home directory. You cannot use the -jdk and -jre options together.
oh	Specifies the Oracle home directory to use instead of the default directory.

rollback Command

The `rollback` command removes a specific interim patch from the appropriate Oracle home directory. The following syntax is used for this command:

```
<Path_to_OPatch>/opatch rollback -id (patch_id) [-ph (patch directory)] \
[-delay (value) [-invPtrLoc (path)] [-jdk (location)] [-jre (location)] \
[-local] [-oh (Oracle Home location)] \
[-post (options to be passed into post) [-opatch_post_end]] \
[-pre (options to be passed into pre) [-opatch_pre_end]] [-retry (value)] \
[-silent] [-verbose] [-no_relink] [-no_sysmod] [-remote_nodes
(node1,node2)] [-local_node (node_name)]
```

Table 4–7 describes the options available for the `rollback` command.

Table 4–7 rollback Command Options

Option	Description
delay	Specifies how many seconds the OPatch utility should wait before attempting to lock the inventory again, if you use the <code>-retry</code> option with the <code>apply</code> command.
id	Indicates the patch to be rolled back. Use the <code>-lsinventory</code> command to display all patch identifiers. To successfully roll back a patch, you must supply the patch identifier.
invPtrLoc	Specifies the location of the <code>oraInst.loc</code> file. This option is needed when the <code>-invPtrLoc</code> option is used during installation. Oracle recommends using the default Central Inventory for a platform.
jdk	Specifies the location of a particular JDK (jar) to use instead of the default location under the Oracle home directory.
jre	Specifies the location of a particular JRE (Java) to use instead of the default location under the Oracle home directory.
local	Specifies that the OPatch utility should roll back and update the local node and update the inventory of the local node. It does not propagate the patch or inventory update to other nodes. You can use this option on Oracle Real Application Clusters environments and non-clustered environments. If you shut down an entire cluster before patching, you can use this argument for non-rolling patches.

Table 4–7 (Cont.) rollback Command Options

Option	Description
local_node	Specifies to the OPatch utility that this is the local node for the cluster. You can use this option on Oracle Real Application Clusters environments.
no_sysmod	Specifies that the OPatch utility need not update the files in the system. It will only update the inventory.
no_relink	This option does not perform any make operation in the patch. You can use it during multiple patch removals and to perform the compilation step only once.
oh	Specifies the Oracle home directory to use instead of the default directory.
opatch_post_end	Marks the end of the <code>post</code> option. This option is used with the <code>post</code> option. If you do not use this argument, everything after <code>post</code> is passed into <code>post</code> .
opatch_pre_end	Marks the end of the <code>pre</code> options. This option is used with the <code>pre</code> option. If you do not use this argument, everything after <code>pre</code> is passed into <code>pre</code> .
ph	Specifies the valid patch directory area. The utility uses the command types found in the patch directory to identify which commands are used for the current operating system.
post	Specifies the parameters to be passed inside the <code>post</code> script besides the standard parameters.
pre	Specifies the parameters to be passed inside the <code>pre</code> script besides the standard parameters.
remote_nodes	Specifies to the OPatch utility the list of remote nodes. You can use this option on Oracle Real Application Clusters environments.
report	Prints the action to the screen without executing it.
retry	Specifies how many times the OPatch utility should try in case of an inventory lock failure.
silent	Suppresses user interaction and defaults any answers to "yes". The Oracle Real Application Clusters setup does not support this option.
verbose	Prints output to the screen as well as to the log file.

version Command

The `version` command shows the current version number of the OPatch utility. The following syntax is used for this command:

```
<Path_to_OPatch>/opatch version
```

Oracle Real Application Clusters Patching

An Oracle Real Application Clusters (RAC) environment is one in which active instances can concurrently execute transactions on a shared database. Patching in an Oracle (RAC) environment is slightly different compared to patching a single node.

Real Application Clusters can be patched in three different ways:

- All Node Patching
- Rolling Patching
- Minimum Downtime Patching

This chapter includes the following topics about patching Oracle RAC:

- [Specifying Node Names and Lists](#)
- [All Node Patching](#)
- [Rolling Patching](#)
- [Minimum Downtime Patching](#)

Specifying Node Names and Lists

Interim patching using OPatch follows a similar approach as that performed by Oracle Universal Installer to detect an Oracle home and nodes of a cluster. OPatch interacts with the Oracle Universal Installer inventory through the Oracle Universal Installer Java SDK. If OPatch detects a cluster, it queries the inventory through Oracle Universal Installer to find the local node name and node list. If your node list is not updated, you can update it by using the `-updateNodeList` flag of Oracle Universal Installer:

```
$ORACLE_HOME/oui/bin/<runInstaller or setup.exe> -updateNodeList  
ORACLE_HOME=<oracle home location> "CLUSTER_NODES={<node1,node2,node3>}"  
-noClusterEnabled
```

You can bypass remote actions using the `-local` flag. If you want to specify the local node or remote nodes of a Real Application Clusters setup to OPatch, you can use the `LOCAL_NODE` or `REMOTE_NODES` session variable and specify the node name(s):

```
$ORACLE_HOME/oui/bin/<runInstaller or setup.exe> ORACLE_HOME=<oracle home  
location>  
"REMOTE_NODES {<node1,node2,node3>}" LOCAL_NODE=<nodelist  
for example:node1>
```

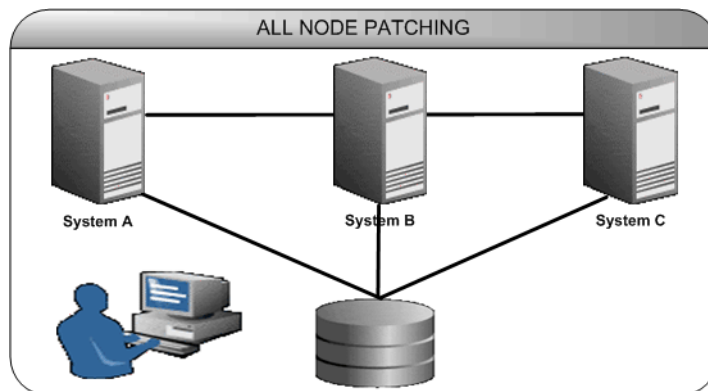
If OPatch does not automatically detect an Oracle RAC or its nodes, you need to investigate the contents of the inventory and ensure that it is complete.

All Node Patching

In All Node Patching, all Oracle RAC nodes are initially brought down and the patch is applied on all the nodes, then all the nodes are brought back up. This mode is normally used for very critical patches and it leads to maximum downtime. OPatch uses this mode as the default for patch applications unless specified otherwise.

Figure 5–1 shows an example of All Node Patching. In this example, Systems A,B, and C are nodes in this Oracle RAC. When you perform an All Node Patching in this cluster, you bring down systems A,B, and C and apply patches to all of these nodes. You then bring up systems A,B, and C again.

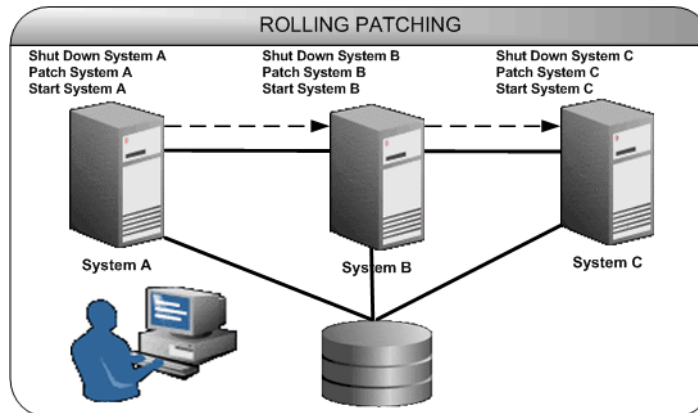
Figure 5–1 All Node Patching



Rolling Patching

In Rolling Patching, each node is shut down, the patch is applied, then each node is brought back up again. This is done node by node separately until all nodes in Oracle RAC are patched. This is the most efficient mode of applying an interim patch to an Oracle RAC setup because this results in no downtime. Only some patches can be applied in this mode. The type is generally specified in the patch metadata.

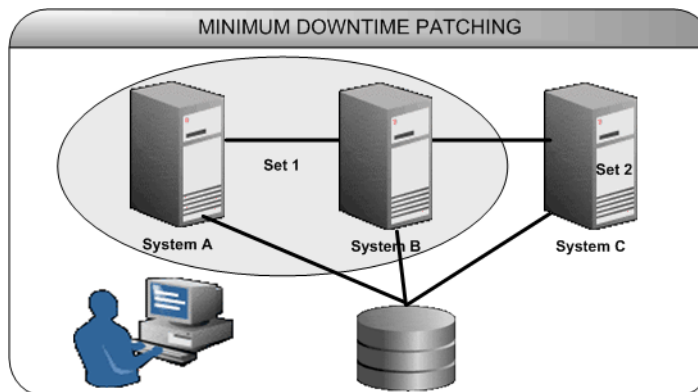
Figure 5–2 shows an example of Rolling Patching. In this example, systems A,B, and C are nodes in this Oracle RAC. When you perform a Rolling Patching in this cluster, you bring down system A, and apply a patch to it, then bring it back up. You then do the same with systems B and C. The patch is thereby applied in a rolling fashion. The main advantage of this type of patching is that there is absolutely no downtime while applying patches because only one system is brought down at any given time.

Figure 5–2 Rolling Patching

Minimum Downtime Patching

In Minimum Downtime Patching, the nodes are divided into sets. Initially, the first set is shut down and the patch is applied to it. After this, the second set is shut down. The first set is brought up and patch is applied to the second set. The second set is also brought up now. All the nodes in the Real Application Clusters are now patched. This mode leads to less downtime for the Real Application Clusters when both the sets are brought down. This mode is executed by using `-minimize_downtime` command line option. You can also activate this option from the response file.

Figure 5–3 shows an example of Minimum Downtime Patching. In this example, systems A,B, and C are nodes in this Oracle RAC. They are divided into two sets: set 1 contains systems A and B, and set 2 contains system C. When you perform Minimum Downtime Patching in this cluster, you shut down set 1 and apply a patch to it. You then shut down set 2, bring up set 1, and apply a patch to set 2. After applying the patch, you bring up set 2 again. Now both sets 1 and 2 are patched.

Figure 5–3 Minimum Downtime Patching

Resolving Patch Conflicts

All patches may not be compatible with one another. For example, if a patch has been applied, all the bugs fixed by that patch could reappear after another patch is applied. This is called a conflict situation. OPatch detects such situations and raises an error when it detects a conflict. This chapter includes the following topics:

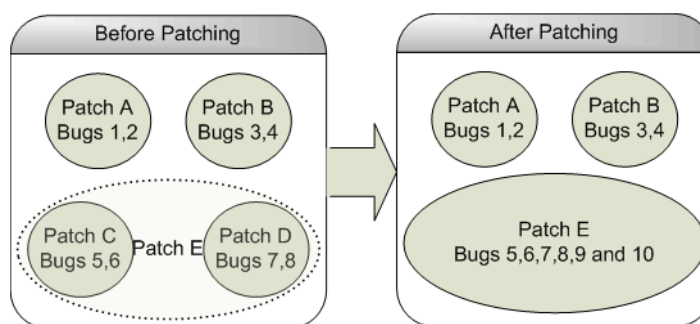
Types of Conflicts

OPatch can detect the following types of conflicts:

- Bug Superset** — If all the bugs fixed by a patch in the system are also fixed by the patch to be applied, this patch (the patch to be applied) is considered to be a superset of the patch already applied. If a bug superset condition is detected, it is not considered an error situation. All the subset patches are removed from the system and the new patch is applied.

For example, consider a scenario where four patches A,B,C, and D are applied in a system, each of which fixes 2 bugs as shown in [Figure 6-1](#). If you apply a patch E that fixes bugs 5,6,7,8,9, and 10, patch E will be the superset of patch C and D.

Figure 6-1 Bug Superset



If you want OPatch to error out if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory, you can use the `-no_bug_superset` flag:

```
$ OPatch/opatch apply -no_bug_superset <Path_To_Patch>
```

The following sample output shows the message you would see when you use the `-no_bug_superset` flag:

```
Oracle interim Patch Installer version 10.2.0.4.6
Copyright (c) 2009, Oracle Corporation. All rights reserved..
```

```

Oracle Home      : /home/oracle_TEST/product/10.2.0/db_1
Central Inventory : /home/OUIHome_Opatch
from            : /home/oracle_TEST/product/10.2.0/db_1/oraInst.loc
OPatch version  : 10.2.0.4.6
OUI version     : 10.2.0.4.6
OUI location    : /home/oracle_TEST/product/10.2.0/db_1/oui
Log file location : /home/oracle_TEST/product/10.2.0/db
_1/cfgtoollogs/opatch/opatch-2008_May_25_14-03-33-IST_Wed.log
ApplySession applying interim patch '102000' to OH '/home/oracle
_TEST/product/10.2.0/db_1'
Apply Session failed: ApplySession failed to prepare the system. Interim patch
102000 is a superset of the patch(es) [ 102000 ] in OH /home/oracle
_TEST/product/10.2.0/db_1
System intact, OPatch will not attempt to restore the system
OPatch failed with error code 73

```

- **Subset** — Patches to be applied can be subsets of other patches installed in the Oracle home.

Consider the following scenario:

- Patch A, installed in the Oracle home, fixed bugs 1, 2, and 3.
- Patch B, installed in the Oracle home, fixed bugs 10, 11, and 12.
- Patch D, to be installed, fixes bugs 1 and 2.

Patch D is a subset of Patch A.

Using the `skip_subset` Option

When you want to skip patches formerly applied in the Oracle home that are now subsets of other patches you want to apply now, you can use the `skip_subset` option of `napply`. For example, if you used `napply` yesterday for patch A that fixed bugs 1 and 2, then you use `napply` today with the `skip_subset` option for patch B that fixes bug 1 and patch C that fixes bugs 1, 2, and 3, then subset patch A is skipped, and patch C then becomes a superset of patch A.

The following example applies all patches under the `<patch_location>` directory. OPatch skips duplicate patches and subset patches (patches under `<patch_location>` that are subsets of patches installed in the Oracle home).

```
opatch napply <patch_location> -skip_subset -skip_duplicate
```

The following example applies patches 1, 2, and 3 that are under the `<patch_location>` directory. OPatch skips duplicate patches and subset patches (patches under `<patch_location>` that are subsets of patches installed in the Oracle home).

```
opatch napply <patch_location> -id 1,2,3 -skip_subset -skip_duplicate
```

See the description for the `skip_subset` option in [Table 4–3](#) for more information.

- **Duplicate** — A duplicate patch fixes the same set of bugs that another patch fixed. For example, if you applied Patch A that fixed bugs 1, 2 and 3, and now apply Patch B that also fixes bugs 1, 2 and 3, then Patch B is a duplicate of Patch A. A patch is always a duplicate of itself.

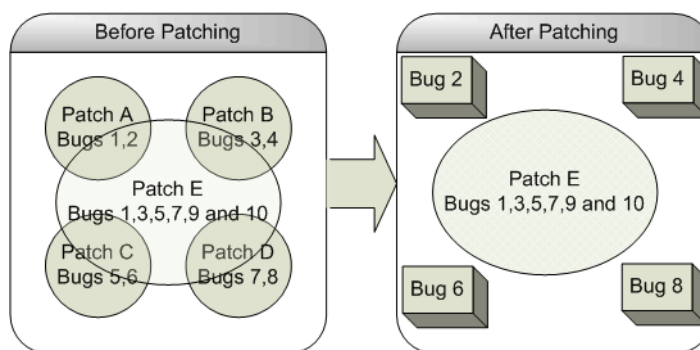
Using the `skip_duplicate` Option

If you specify this option, OPatch removes duplicate patches from the list of patches to be applied. For example, if you used `napply` yesterday for Patch A discussed above, then use `napply` today with the `-skip_duplicate` option for Patch A and other patches, duplicate Patch A is skipped.

- **Bug Conflict** — If a set of bugs to be fixed by the current interim patch includes some but not all bugs already fixed by one or more previously installed interim patches, this is considered a bug conflict. You must remove the bug conflict before proceeding with the patch by using the `apply` command with the `-force` flag, which rolls back the conflicting patches before applying the new one.

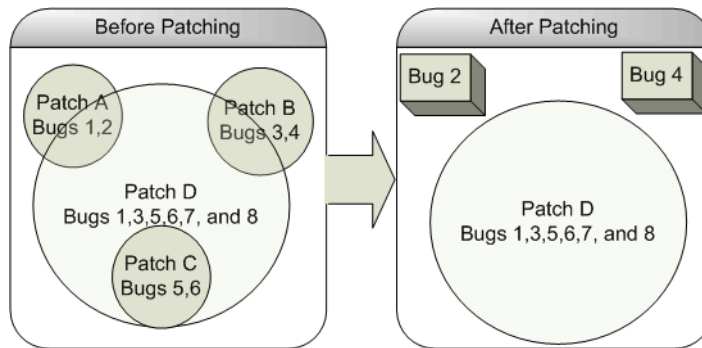
For example, consider a scenario where four patches A,B,C, and D are applied in a system, each of which fixes 2 bugs as shown in [Figure 6-2](#). If you apply a patch E that fixes bugs 1,3,5,7,9, and 10, this patch fixes bugs 1,3,5,7,9, and 10, opens bugs 2,4,6, and, 8. This is a conflict situation.

Figure 6-2 Bug Conflict



- **File Conflict** — If a set of files to be patched by the current interim patch include files already patched by one or more previously installed interim patches and it is not a bug superset, this is considered a file conflict. You must remove the file conflict before you proceed with patching by using the `apply` command with the `-force` flag, which rolls back the conflicting patches before applying the new one.
- **Combination Conflict** — If a set of patches has a combination of a bug superset and bug or file conflict, this is considered a combination conflict and is an error situation. In this case, OPatch removes all conflicting patches as well as the subset patches and then re-applies the new patch.

For example, consider a scenario where four patches A,B,C, and D are applied in a system, each of which fixes 2 bugs as shown in [Figure 6-3](#). Patch C is the subset of patch D. Patch A and patch B are conflicting patches of patch D. If you apply this patch D that fixes bugs 1,3,5,6,7, and 8 with the `-force` flag, OPatch rolls back patches A, B, and C and applies patch D.

Figure 6–3 Combination Conflict

You can use the `-silent`, `-force`, and `-no_bug_superset` options in Real Application Clusters. Their expected behavior is described in [Table 6–1](#).

Table 6–1 Truth Table

<code>-silent</code>	<code>-force</code>	<code>-no_bug_superset</code>	Expected behavior
N	N	N	OPatch prompts questions and applies the patches depending upon your responses.
N	N	Y	OPatch errors out if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory. If it is a conflict case, OPatch prompts questions and applies the patches depending upon your responses.
N	Y	N	OPatch prompts questions and applies the patches depending upon your responses.
N	Y	Y	OPatch errors out if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory. If it is a conflict case, OPatch prompts questions and applies the patches depending upon your responses.
Y	N	N	OPatch rolls back and applies the patch if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory. If it is a conflict case, OPatch errors out.
Y	N	Y	OPatch errors out if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory. If it is a conflict case also, OPatch errors out.
Y	Y	N	OPatch rolls back all the patches and reapplies them without warning, even if the current patch bugs-to-fix is a superset, or a conflict, or the same as an installed patch bugs-fixed in the Oracle home directory.
Y	Y	Y	OPatch errors out if the current patch bugs-to-fix is a superset or the same as an installed patch bugs-fixed in the Oracle home directory. If it is a conflict case, OPatch rolls back and applies the new patch.

Patch Conflict Behavior for Apply and Napply

The expected behavior for the Apply and Napply commands is listed in [Table 6–2](#).

Table 6–2 *Expected Behavior for Apply and Napply Commands*

Command	Superset	Subset	Duplicate	File Conflict or Bug Conflict Patch
Apply	OPatch performs an automatic rollback, then an apply.	After the merge request, OPatch performs an automatic rollback, then performs an apply.	OPatch performs an automatic rollback, then performs a reapply.	OPatch reports the conflict. After the merge request, OPatch performs an automatic rollback, then an apply.
Napply	OPatch performs an automatic rollback, then an apply.	OPatch reports the subset and skips the subset patch. It then continues and applies the other patches.	OPatch performs an automatic rollback, then a reapply.	OPatch reports the conflict, then asks you to run again without applying a bug conflict patch. You can use the <code>-force</code> option to instruct OPatch to automatically roll back the conflicting patch, then apply the new patch.

Patch Conflict Detection and Resolution

OPatch detects and reports any conflicts encountered when applying an interim patch with a previously applied patch. The patch application fails in case of conflicts. You can use the `-force` option of OPatch to override this failure. If you specify `-force`, the installer firsts rolls back any conflicting patches and then proceeds with the installation of the desired interim patch.

You may experience a bug conflict and might want to remove the conflicting patch. This process is known as patch rollback. During patch installation, OPatch saves copies of all the files that were replaced by the new patch before the new versions of these files are loaded, and stores them in `$ORACLE_HOME/.patch_storage`. These saved files are called rollback files and are key to making patch rollback possible. When you roll back a patch, these rollback files are restored to the system. If you have gained a complete understanding of the patch rollback process, you should only override the default behavior by using the `-force` flag. To roll back a patch, execute the following command:

```
$ OPatch/opatch rollback -id <Patch_ID>
```

Recovery and Troubleshooting

The OPatch utility is a tool that allows the application and rollback of interim patches to Oracle products. This chapter provides information on using OPatch to apply patches.

This chapter includes the following topics:

- [Logging and Tracing](#)
- [Restoring Oracle Homes](#)
- [Recovering from a Failed Patching Session Using OPatch](#)
- [Troubleshooting OPatch](#)

Logging and Tracing

Logging and tracing is a common aid for debugging. OPatch maintains logs for all apply, rollback, and lsInventory operations. The log files are located in the <ORACLE_HOME>/cfgtoollogs/opatch directory. Each log file is tagged with the timestamp of the operation. Log files are named as opatch_<date mm-dd-yyyy>_<time hh-mm-ss>.log. A new log file is created each time OPatch is executed.

For example, if a log file is created on May 17th, 2009 at 11.55 PM, it will be named as follows:

```
opatch_05-17-2009_23-55-00.log
```

Note: You can set OPatch to debug mode by setting the environment variable OPATCH_DEBUG to TRUE.

OPatch also maintains an index of the commands executed with OPatch and the log files associated with it in the `history.txt` file located in the <ORACLE_HOME>/cfgtoollogs/opatch directory. A sample of the `history.txt` file is as follows:

```
Date & Time : Tue Apr 26 23:00:55 PDT 2008
Oracle Home : /private/oracle/product/10.2.0/db_1/
OPatch Ver. : 10.2.0.4.6
Current Dir : /scratch/oui/OPatch
Command    : lsinventory
Log File   :
/private/oracle/product/10.2.0/db_1/cfgtoollogs/opatch/opatch-2008_Apr_26_
23-00-55-PDT_Tue.log
```

OPatch follows the Oracle Diagnostic Logging (ODL) Guidelines. You can set the log level by using the available `-logLevel <level>` option. This option controls the amount of logging OPatch performs according to the ODL guidelines.

OPatch supports the following log levels:

- SEVERE
- WARNING
- INFO
- CONFIG
- FINE
- FINER
- FINEST

Restoring Oracle Homes

Every time you apply a patch, you make changes to your inventory. Sometimes that change may corrupt the inventory. You can use the `restore.sh` or `restore.bat` script that comes with OPatch to remove any changes that were made to the inventory after the patch application.

When you apply a patch, OPatch creates a snapshot of your inventory and stores it in `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory. OPatch 10.2 supports maintaining versions of patches. You can have two or more different versions of the same patch (with the same patch ID). This version information is stored in the OPatch metadata.

If your inventory is corrupted, you need to perform the following steps to return the application to its last known good state:

1. Ensure that the environment variable `ORACLE_HOME` is set properly.
2. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command:

```
For UNIX:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

```
For Windows:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt` file (if available) as follows:

```
/bin/sh make.txt
```

Recovering from a Failed Patching Session Using OPatch

During patching, updates can occur in two phases:

- **System Update** — In this phase, the files are replaced in the Oracle home.
- **Inventory Update** — In this phase, the details of the patch applied is recorded in the inventory.

The following scenarios explain how can you recover from a failed patching session:

The ORACLE_HOME does not appear when you execute `opatch lsinventory -detail`.

Cause: The ORACLE_HOME might be missing from the Central Inventory or the Central Inventory itself could be missing or corrupted.

Action: Ensure that the ORACLE_HOME is in the Central Inventory and/or the Central Inventory is not missing or corrupted.

The ORACLE_HOME is listed when you execute `opatch lsinventory -detail`, **but the products and components within the ORACLE_HOME are not listed.**

Cause: The inventory within the ORACLE_HOME (local inventory) could be missing or corrupted.

Action: If the local inventory is corrupted or lost for any reason, just restore the ORACLE_HOME/inventory if it had been backed up. If a back-up does not exist, you may have to reinstall the software.

When you apply or rollback a patch, you receive an interim inventory update exception.

Cause: This occurs when the files on the system are patched, but the inventory update has failed. This may be due to a corrupted inventory.

Action: Perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly.
2. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command:

```
For UNIX:
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

```
For Windows:
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt` file (if available) as follows:

```
/bin/sh make.txt
```

When you apply or rollback a patch, you receive an OiiOneoffException exception.

Cause: This occurs when the files on the system are patched, but the inventory update has failed. This may be because the base component of the interim patch may not be present in the inventory.

Action: OPatch attempts to restore the Oracle home automatically and provides a message for this action. If OPatch does not provide a message stating that it has restored the Oracle home, perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly.
2. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command:

```
For UNIX:
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

```
For Windows:
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt` file (if available) as follows:

```
/bin/sh make.txt
```

When you apply a patch and execute `opatch lsinventory`, it does not return anything.

Cause: This may be due to losing all the patches applied before the application of the current patch, or the patches might not have been updated in the inventory.

Action: Perform the following steps:

1. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command:

For UNIX:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

For Windows:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

2. On UNIX, source the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt` file (if available) as follows:

```
/bin/sh make.txt
```

3. If the files are properly patched, but the information is not updated in the inventory, execute the following command:

```
$ORACLE_HOME/OPatch/opatch apply -no_sysmod <Path_To_Patch>
```

Ensure that the patch has been applied and recorded properly in the inventory by executing the following command:

```
$ORACLE_HOME/OPatch/opatch lsinventory -detail
```

4. If the files are still not patched properly, but you are able to see the patch in the `lsinventory` flag, you need to reapply the patch using the `no_inventory` flag:

```
$ORACLE_HOME/OPatch/opatch apply -no_inventory <Path_To_Patch>
```

When you apply a patch and execute `opatch lsinventory`, it does not return the details of the patch applied.

Cause: This may be because OPatch may not have recorded the details of this patch in the inventory.

Action: Perform the following steps:

1. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command:

For UNIX:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

For Windows:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

2. On UNIX, source the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt` file (if available) as follows:

```
/bin/sh make.txt
```

3. If the files are properly patched, but the information is not updated in the inventory, execute the following command:

```
$ORACLE_HOME/OPatch/opatch apply -no_sysmod <Path_To_Patch>
```

Ensure that the patch has been applied and recorded properly in the inventory by executing the following command:

```
$ORACLE_HOME/OPatch/opatch lsinventory -detail
```

When you press Ctrl + C during the application or rollback of a patch and execute `opatch lsinventory`, it does not return the details of the patch applied or rolled back.

Cause: This may be because OPatch might have stopped the application or rollback of the patch upon pressing Ctrl + C.

Action: Perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly.
2. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command if it is available:

For UNIX:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

For Windows:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt` file (if available) as follows:

```
/bin/sh make.txt
```

When you apply a patch, the process quit when OPatch failed to relink and prompted to continue.

Cause: This may be due to relink failure.

Action: Perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly.
2. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command.

For UNIX:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

For Windows:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. Resolve the re-link failure issue by ensuring that you are able to invoke make manually on a UNIX shell. After this, apply the patch again.

Oracle Real Application Clusters (RAC) Setup

When I apply a patch on an Oracle RAC setup, and execute 'opatch lsinventory' on the local node, the patch is not listed.

Cause: This may occur if OPatch failed to update the inventory.

Action: Perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly in all the nodes of the cluster.
2. Navigate to the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp> directory of each node in the cluster and execute the restore command as follows:

```
For UNIX:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

```
For Windows:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt file (if available) in each node of the cluster as follows:

```
/bin/sh make.txt
```

4. Apply the patch in each node in the cluster using the local flag:

```
$ORACLE_HOME/OPatch/opatch apply -local <Path_To_Patch>
```

Note: Ensure that all the nodes use the same OPatch version.

When I apply a patch on an Oracle RAC setup and execute 'opatch lsinventory' on the local node, it does not return anything.

Cause: You might have lost all the patches applied earlier.

Action: Perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly in each node in the cluster.
2. Navigate to the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp> directory and execute the restore command in each node in the cluster.

```
For UNIX:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

```
For Windows:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt file (if available) in each node as follows:

```
/bin/sh make.txt
```

4. Apply the patch in each node using the local flag:

```
$ORACLE_HOME/OPatch/opatch apply -local <Path_To_Patch>
```

Note: Ensure that all the nodes use the same OPatch version.

When I rollback a patch on an Oracle RAC setup and execute 'opatch lsinventory' on the local node, it shows that the patch was not removed.

Cause: This may occur if OPatch failed to update the inventory.

Action: Perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly in each node in the cluster.
2. Navigate to the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp> directory in each node in the cluster and execute the restore command as follows:

For UNIX:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

For Windows:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt file (if available) in each node in the cluster as follows:

```
/bin/sh make.txt
```

4. Roll back the patch in all the nodes in the cluster using the local flag:

```
$ORACLE_HOME/OPatch/opatch rollback -local -id <Patch_ID>
```

Note: Ensure that all the nodes use the same OPatch version.

When I roll back a patch on an Oracle RAC setup and execute 'opatch lsinventory' on the local node, it does not return anything.

Cause: You might have lost all the patches applied earlier.

Action: Perform the following steps:

1. Ensure that the environment variable ORACLE_HOME is set properly in each node in the cluster.
2. Navigate to the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp> directory and execute the restore command in each node in the cluster:

For UNIX:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

For Windows:

```
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt file (if available) as follows:

```
/bin/sh make.txt
```

4. Roll back the patch in the local node using the local flag:

```
$ORACLE_HOME/OPatch/opatch rollback -local -id <Patch_ID>
```

5. Roll back the patch on the other nodes also using local flag.

Note: Ensure that all the nodes use the same OPatch version.

When I apply a patch on an Oracle RAC setup, the patching in one node is fine (both the files and the inventory are fine), but I am not sure about the other nodes.

Cause: This may be due to a failed system or inventory update.

Action: Perform the following steps:

1. Copy the Oracle home from the node that is fine to the other nodes.
2. After copying the Oracle home, make sure that the `ORACLE_HOME/inventory/ContentsXML/comps.xml` file has the latest time stamp.

Note: On Unix, use `touch` to change the time stamp.

3. Update the nodes of the cluster. For more information on updating the nodes of the cluster, refer to the *Oracle Universal Installer User's Guide*.
4. Ensure that all the prerequisite checks listed in [Chapter 3, "OPatch Prerequisite Checks"](#) pass.

When I apply a patch on an Oracle RAC setup, the patching in one node is fine, but when I execute 'opatch lsinventory' on the other nodes, the patch is not listed.

Cause: This may be due to a failed system or inventory update.

Action: Perform the following steps:

1. Copy the `ORACLE_HOME /inventory` directory from the node that is fine to the other nodes.
2. After copying the `ORACLE_HOME /inventory` directory, make sure that the `ORACLE_HOME/inventory/ContentsXML/comps.xml` file has the latest time stamp.

Note: On Unix, use `touch` to change the time stamp.

3. Update the nodes of the cluster. For more information on updating the nodes of the cluster, see the *Oracle Universal Installer User's Guide*.
4. Ensure that all the prerequisite checks listed in the section [Chapter 3, "OPatch Prerequisite Checks"](#) pass.

When I apply or roll back a patch on an Oracle RAC setup, I am not able to apply or roll back the patch on all nodes.

Cause: This may occur if the nodes are not properly updated.

Action: Perform any one or more of the following:

- Ensure that all the nodes in the cluster are up-to-date. If they are not, update the nodes of the cluster. For more information on updating the nodes of the cluster, see the *Oracle Universal Installer User's Guide*.
- Execute the appropriate command on all nodes of the cluster as follows:


```
opatch apply -local [patch_location]
opatch rollback -local [patch_location]
```
- Execute the appropriate command on the local node of the cluster as follows:


```
opatch apply [-local_node (node_name)] [-remote_nodes (comma-separated node_names)]
opatch rollback [-local_node (node_name)] [-remote_nodes (comma-separated
```

```
node_names)]
```

Troubleshooting OPatch

This section provides solutions to errors that may occur during patch application.

- [Not a valid patch area](#)
- [OPatch cannot find system commands like fuser and make](#)
- [Unable to remove a partially-installed interim patch](#)

Not a valid patch area

Cause: The directory that the OPatch utility is using to do the patch does not match the template for what it is checking. This can also occur when the utility is run from an invalid shiphome directory.

Action: When starting the OPatch utility, the directory needs the following:

- `/etc` directory that has the metadata files.
- `/files` directory that has the payload files.
- The `/etc/config/inventory` file and the actions file under the same directory.

If you did not start the OPatch utility from the `patch_id` directory, you can use the following command:

```
opatch apply /<Patch_Shiphome>
```

OPatch cannot find system commands like fuser and make

Cause: The OPatch utility uses `fuser` on UNIX systems to check for active Oracle instances. On certain hp-ux systems, only a super-user can run `fuser`.

Action: Perform the following steps to resolve this problem:

1. Set `/tmp` in your PATH.

For more information, refer to "[Check for System Commands](#)" on page 3-2.

2. Create an empty file named `fuser`.
3. Shut down the Oracle instances.
4. Run the OPatch utility.

Caution: Another way to resolve this problem is to give executable permission to other users for `fuser`. However, this exposes a potential security risk in the system and is not recommended.

Unable to remove a partially-installed interim patch

Cause: Interruption in the patching process is the potential cause for this problem. This may occur if you press CTRL+C during the patching process. If the error is the one that OPatch detects, it automatically resolves it.

Action: Perform the following steps:

1. Ensure that the environment variable `ORACLE_HOME` is set properly.
2. Navigate to the `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>` directory and execute the restore command as follows:

```
For UNIX:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.sh
```

```
For Windows:  
$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/restore.bat
```

3. On UNIX, source the \$ORACLE_HOME/.patch_storage/<patch-id_timestamp>/make.txt file (if available) as follows:

```
/bin/sh make.txt
```

Frequently Asked Questions

This section answers the frequently asked questions for OPatch.

When I apply a patch I get an error that reads "Failed to load the patch object." Possible causes are: The specified path is not an interim patch shiphome. Meta-data files are missing from the patch area ". What do I do?

This simply means the directory OPatch is using to find the patch doesn't match the template it is checking for. For more information on this error, see "Not a valid patch area" in section "[Troubleshooting OPatch](#)" on page 7-9.

When I apply a patch I get an error that reads "Syntax error.....Patch location not valid". What do I do?

This simply means that the patch location that you specify is an invalid one. Give the correct patch location and apply the patch again.

When I apply a patch I get an error that reads "Exception in thread "main" java.lang.NoClassDefFoundError: <Class Name>". What do I do?

This may be due to OPatch not able to find the particular class listed in the error, which is supposed to be located inside \$ORACLE_HOME/OPatch/jlib/opatch.jar file. Check if you have the particular class file there. To check this, execute the following command; the missing class file will be printed out:

```
cd $ORACLE_HOME/OPatch/jlib  
jar tf opatch.jar <Class File Name>.class
```

It is recommended that you contact Oracle support when you encounter this error.

Another reason might be having done a file transfer of OPatch in a non-binary mode.

When I apply a patch, I get an error that reads "OPatch cannot find the required command 'ar' from Property file and your PATH". What do I do?

'ar' is a command used by OPatch. This message may appear if OPatch is not able to locate this command.

For more details and work-around for this problem, refer to "OPatch cannot find system commands like fuser, make" in section "[Troubleshooting OPatch](#)" on page 7-9.

When I apply a patch, I get an error that reads "OPatch cannot find the required command 'fuser' from Property file and your PATH". What do I do?

'fuser' is a command used by OPatch. This message may appear if OPatch is not able to locate this command.

For more details and work-around for this problem, see "OPatch cannot find system commands like fuser, make" in section "[Troubleshooting OPatch](#)" on page 7-9.

How do I get the information about a patch that I applied long back?

You can look at the folder `$ORACLE_HOME/.patch_storage/<patch-id_timestamp>`. It has detailed information about the patch. You can also use `opatch lsinventory -detail` to see the files that have been modified by the patch.

Where do I get the OPatch 10.2 log files?

You can look at the folder `$ORACLE_HOME/cfgtoollogs/patch` for OPatch 10.2 log files.

How do I find out a list of Oracle home(s) for a host?

To find out the list of Oracle home(s) in a host, use the command `lsinventory -all`.

How can I minimize the downtime when applying a patch to a Real Application Clusters setup?

You can minimize the downtime when applying a patch to a Real Application Clusters setup by doing Minimum Downtime Patching. For more information refer to section "[Minimum Downtime Patching](#)" on page 5-3.

Can I stop applying a patch after applying it to a few nodes? What are the possible issues?

Yes, it is possible to stop applying a patch after applying it to a few nodes. But, Oracle recommends that you do not do this.

There is a prompt that allows you to stop applying the patch. This means you cannot apply another patch until the process is restarted and all the nodes are patched or the partially applied patch is rolled back.

Can I run patching in scripted mode?

Yes, it is possible by using the command `opatch <option> -silent`. For more information, refer to the `-silent` option documented for several commands in [Chapter 4, "OPatch Utility and Commands"](#).

Before applying a patch I want to know what is the impact of the patch?

You can use the command `opatch <option> -report`. For more information, refer to the `-report` option documented for several commands in [Chapter 4, "OPatch Utility and Commands"](#).

What versions of OPatch can I use with Oracle Universal Installer 10.2?

Oracle recommends using OPatch version 10.2 from the Oracle home with Oracle Universal Installer 10.2. Also note that OPatch is compatible only with the version of Oracle Universal Installer that is installed in the Oracle home.

Is OPatch 10.2 backward compatible? Can I use OPatch 10.2 to apply 9.2 and 10.1 patches?

No, OPatch 10.2 is not backward compatible. You can use OPatch 10.2 only to apply 10.2 patches.

When I apply a patch, I get an error that reads as follows:

"OPatchSession cannot load inventory for the given Oracle Home <Home_Location>. Possible causes are:

No read or write permission to ORACLE_HOME/.patch_storage

Central Inventory is locked by another OUI instance

No read permission to Central Inventory

The lock file exists in ORACLE_HOME/.patch_storage

"The Oracle Home does not exist in Central Inventory"

What do I do?

This error may occur because of any one or more of the following reasons:

1. The ORACLE_HOME/.patch_storage may not have read/write permissions. Ensure that you give read/write permissions to this folder and apply the patch again.
2. There may be another Oracle Universal Installer instance running. Stop it and try applying the patch again.
3. The Central Inventory may not have read permission. Ensure that you give read permission to the Central Inventory and apply the patch again.
4. The ORACLE_HOME/.patch_storage directory might be locked. If this directory is locked, you will find a file named patch_locked inside this directory. This may be due to a previously failed installation of a patch. To remove the lock, restore the Oracle home and remove the patch_locked file from the ORACLE_HOME/.patch_storage directory. For more information on restoring the Oracle home, refer to section "[Restoring Oracle Homes](#)" on page 7-2.
5. The Oracle home may not be present in the Central Inventory. This may be due to a corrupted or lost inventory or the inventory may not be registered in the Central Inventory. For more information, refer to "Diagnosing and Recovering from Central Inventory Corruption" in the *Oracle Universal Installer User's Guide*.