

Oracle Database 11g: Administration Workshop I

Volume II • Student Guide

D50102GC20

Edition 2.0

September 2009

D62542

ORACLE®

Authors

Deirdre Matishak

Mark Fuller

Technical Contributors and Reviewers

Maria Billings

Herbert Bradbury

Yanti Chang

Timothy Chien

Andy Fotunak

Gerlinde Frenzen

Steve Friedberg

Joel Goodman

Vimala Jacob

Dominique Jeunot

Pete Jones

Fukue Kawabe

Donna Keesling

Sean Kim

Achiel Langers

Gwen Lazenby

Essi Parast

Randy Richeson

Joe Roch

Hilda Simon

Ira Singer

Jim Spiller

Supithran Thananayagam

Branislav Valny

Manju Varrier

Editors

Raj Kumar

Daniel Milne

Graphic Designer

Rajiv Chandrabhanu

Publishers

Jobi Varghese

Veena Narasimhan

Copyright © 2009, Oracle. All rights reserved.

Disclaimer

This document contains proprietary information and is protected by copyright and other intellectual property laws. You may copy and print this document solely for your own use in an Oracle training course. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice. If you find any problems in the document, please report them in writing to: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. This document is not warranted to be error-free.

Restricted Rights Notice

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

Trademark Notice

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

Contents

I Introduction

Course Objectives	I-2
Suggested Schedule	I-3
Oracle Products and Services	I-4
Oracle Database 11g: "g" Stands for Grid	I-5
Grid Infrastructure for Single-Instance	I-7

1 Exploring the Oracle Database Architecture

Objectives	1-2
Oracle Database	1-3
Connecting to a Server	1-4
Oracle Database Server Architecture: Overview	1-6
Instance: Database Configurations	1-7
Connecting to the Database Instance	1-8
Oracle Database Memory Structures	1-9
Shared Pool	1-11
Database Buffer Cache	1-13
Redo Log Buffer	1-14
Large Pool	1-15
Java Pool and Streams Pool	1-16
Program Global Area (PGA)	1-17
Quiz	1-18
Process Architecture	1-20
Process Structures	1-21
Database Writer Process (DBW n)	1-23
Log Writer Process (LGWR)	1-25
Checkpoint Process (CKPT)	1-27
System Monitor Process (SMON)	1-28
Process Monitor Process (PMON)	1-29
Recoverer Process	1-30
Archiver Processes (ARC n)	1-31
Process Startup Sequence	1-32
Database Storage Architecture	1-33
Logical and Physical Database Structures	1-35

Segments, Extents, and Blocks 1-37
Tablespaces and Data Files 1-38
SYSTEM and SYSAUX Tablespaces 1-39
Automatic Storage Management 1-40
ASM Storage Components 1-41
Interacting with an Oracle Database: Memory, Processes and Storage 1-42
Quiz 1-44
Summary 1-46
Practice 1: Overview 1-47

2 Installing your Oracle Software

Objectives 2-2
Tasks of an Oracle Database Administrator 2-3
Tools for Administering an Oracle Database 2-4
Planning Your Installation 2-6
Oracle Grid Infrastructure and Oracle Database Installation: System Requirements 2-8
Preparing the Operating System 2-9
Setting Environment Variables 2-10
Checking the System Requirements 2-11
Oracle Universal Installer (OUI) 2-12
Example: Installation Scenario 2-13
Part One: Installing the Oracle Grid Infrastructure for Standalone Server 2-14
Selecting Product Languages 2-15
Creating an ASM Disk Group 2-16
Defining ASM Passwords 2-17
Defining Privileged Operating System Groups 2-18
Specifying Installation Location 2-19
Creating Inventory 2-20
Performing Prerequisite Checks 2-21
Verifying Installation Summary Data 2-22
Monitoring Installation Progress 2-23
Executing `root` Configuration Scripts 2-24
Executing Configuration Assistants 2-25
Finishing Installation 2-26
Configuring the FRA Disk Group 2-27
Quiz 2-28
Part Two: Installing the Oracle Database Software 2-30
Choosing the Type of Installation 2-31
Choosing Grid Installation Options 2-32
Choosing Language Settings 2-33

Choosing the Database Edition	2-34
Specifying Installation Location	2-35
Choosing Operating System Groups	2-36
Performing Prerequisite Checks	2-37
Installation Summary Page	2-38
The Install Product Page	2-39
Installation Finished	2-40
Installation Option: Silent Mode	2-41
Quiz	2-42
Summary	2-44
Practice 2 Overview: Preparing the Database Environment	2-45

3 Creating an Oracle Database Using DBCA

Objectives	3-2
Planning the Database	3-3
Databases: Examples	3-4
Choosing the Appropriate Character Set	3-5
How Are Character Sets Used?	3-7
Problems to Avoid	3-8
Database Configuration Assistant (DBCA)	3-9
Using the DBCA to Create a Database	3-10
Create Database Summary	3-16
Password Management	3-17
Creating a Database Design Template	3-18
Using the DBCA to Delete a Database	3-19
Using the DBCA for Additional Tasks	3-21
Quiz	3-22
Summary	3-24
Practice 3 Overview: Using the DBCA	3-25

4 Managing the Database Instance

Objectives	4-2
Management Framework	4-3
Starting and Stopping Database Control	4-4
Oracle Enterprise Manager	4-5
Database Home Page	4-7
Other Oracle Tools	4-8
Using SQL*Plus	4-9
Calling SQL*Plus from a Shell Script	4-10
Calling a SQL Script from SQL*Plus	4-11
Initialization Parameter Files	4-12

Simplified Initialization Parameters 4-14
Initialization Parameters: Examples 4-15
Using SQL*Plus to View Parameters 4-19
Changing Initialization Parameter Values 4-21
Changing Parameter Values: Examples 4-23
Quiz 4-24
Database Startup and Shutdown: Credentials 4-26
Starting Up an Oracle Database Instance 4-27
Starting Up an Oracle Database Instance: NOMOUNT 4-28
Starting Up an Oracle Database Instance: MOUNT 4-29
Starting Up an Oracle Database Instance: OPEN 4-30
Startup Options: Examples 4-31
Shutting Down an Oracle Database Instance 4-32
Shutdown Modes 4-33
Shutdown Options 4-34
Shutdown Options: Examples 4-37
Viewing the Alert Log 4-38
Using Trace Files 4-40
Dynamic Performance Views 4-42
Dynamic Performance Views: Usage Examples 4-43
Dynamic Performance Views: Considerations 4-44
Data Dictionary: Overview 4-45
Data Dictionary Views 4-46
Data Dictionary: Usage Examples 4-48
Quiz 4-49
Summary 4-51
Practice 4 Overview: Managing the Oracle Instance 4-52

5 Managing the ASM Instance

Objectives 5-2
ASM Benefits for Administrators 5-3
ASM Instance 5-4
ASM Components: ASM Instance—Primary Processes 5-6
ASM Instance Initialization Parameters 5-7
Interaction Between Database Instances and ASM 5-9
ASM Instance: Dynamic Performance Views 5-10
ASM System Privileges 5-11
Using Enterprise Manager to Manage ASM Users 5-12
Starting and Stopping ASM Instances Using SQL*Plus 5-13
Starting and Stopping ASM Instances Using `srvctl` 5-15

Starting and Stopping ASM Instances Using <code>asmcmd</code>	5-16
Disk Group Overview	5-17
ASM Disks	5-18
Allocation Units	5-19
ASM Files	5-20
Extent Maps	5-21
Striping Granularity	5-22
Fine-Grained Striping	5-23
ASM Failure Groups	5-25
Stripe and Mirror Example	5-26
Failure Example	5-27
Managing Disk Groups	5-28
Creating and Dropping Disk Groups Using SQL*Plus	5-29
Adding Disks to Disk Groups	5-30
Miscellaneous <code>ALTER</code> Commands	5-31
ASM Management Using Enterprise Manager	5-32
ASM Disk Group Compatibility	5-33
ASM Disk Group Attributes	5-35
Using Enterprise Manager to Edit Disk Group Attributes	5-36
Retrieving ASM Metadata	5-37
ASM Fast Mirror Resync Overview	5-38
Quiz	5-39
Summary	5-41
Practice 5 Overview: Managing the ASM Instance	5-42

6 Configuring the Oracle Network Environment

Objectives	6-2
Oracle Net Services	6-3
Oracle Net Listener	6-4
Establishing Net Connections	6-5
Establishing a Connection	6-6
User Sessions	6-7
Tools for Configuring and Managing the Oracle Network	6-8
Listener Control Utility	6-10
Listener Control Utility Syntax	6-11
Using <code>SRVCTL</code> to Start and Stop the Listener	6-13
Listener Home Page	6-14
Net Services Administration Page	6-15
Creating a Listener	6-16
Adding Listener Addresses	6-17
Database Service Registration	6-18

Naming Methods	6-20
Easy Connect	6-21
Local Naming	6-22
Directory Naming	6-23
External Naming Method	6-24
Configuring Service Aliases	6-25
Advanced Connection Options	6-26
Testing Oracle Net Connectivity	6-28
User Sessions: Dedicated Server Process	6-29
User Sessions: Shared Server Processes	6-30
SGA and PGA	6-31
Shared Server: Connection Pooling	6-32
When Not to Use a Shared Server	6-33
Configuring Communication Between Databases	6-34
Connecting to Another Database	6-35
Quiz	6-36
Summary	6-38
Practice 6 Overview: Working with Oracle Network Components	6-39

7 Managing Database Storage Structures

Objectives	7-2
How Table Data Is Stored	7-3
Database Block: Contents	7-4
Exploring the Storage Structure	7-5
Creating a New Tablespace	7-6
Storage for Tables espaces	7-8
Tablespaces in the Preconfigured Database	7-10
Altering a Tablespace	7-12
Actions with Tables espaces	7-14
Dropping Tables espaces	7-16
Viewing Tablespace Information	7-17
Viewing Tablespace Contents	7-18
Oracle-Managed Files (OMF)	7-19
Enlarging the Database	7-21
Quiz	7-22
Summary	7-24
Practice 7 Overview: Managing Database Storage Structures	7-25

8 Administering User Security

Objectives	8-2
Database User Accounts	8-3

Predefined Administrative Accounts 8-5
Creating a User 8-6
Authenticating Users 8-7
Administrator Authentication 8-9
Unlocking a User Account and Resetting the Password 8-10
Privileges 8-11
System Privileges 8-12
Object Privileges 8-14
Revoking System Privileges with `ADMIN OPTION` 8-15
Revoking Object Privileges with `GRANT OPTION` 8-16
Benefits of Roles 8-17
Assigning Privileges to Roles and Assigning Roles to Users 8-18
Predefined Roles 8-19
Creating a Role 8-20
Secure Roles 8-21
Assigning Roles to Users 8-22
Quiz 8-23
Profiles and Users 8-25
Implementing Password Security Features 8-27
Creating a Password Profile 8-29
Supplied Password Verification Function: `VERIFY_FUNCTION_11G` 8-30
Assigning Quotas to Users 8-31
Applying the Principle of Least Privilege 8-33
Protect Privileged Accounts 8-35
Quiz 8-36
Summary 8-38
Practice 8 Overview: Administering Users 8-39

9 Managing Data Concurrency

Objectives 9-2
Locks 9-3
Locking Mechanism 9-4
Data Concurrency 9-5
DML Locks 9-7
Enqueue Mechanism 9-8
Lock Conflicts 9-9
Possible Causes of Lock Conflicts 9-10
Detecting Lock Conflicts 9-11
Resolving Lock Conflicts 9-12
Resolving Lock Conflicts with SQL 9-13

Deadlocks 9-14
Quiz 9-15
Summary 9-17
Practice 9 Overview: Managing Data and Concurrency 9-18

10 Managing Undo Data

Objectives 10-2
Undo Data 10-3
Transactions and Undo Data 10-5
Storing Undo Information 10-6
Undo Data Versus Redo Data 10-7
Managing Undo 10-8
Configuring Undo Retention 10-9
Guaranteeing Undo Retention 10-11
Changing an Undo Tablespace to a Fixed Size 10-12
General Undo Information 10-13
Using the Undo Advisor 10-14
Viewing System Activity 10-15
Quiz 10-16
Summary 10-18
Practice 10 Overview: Managing Undo Segments 10-19

11 Implementing Oracle Database Auditing

Objectives 11-2
Separation of Responsibilities 11-3
Database Security 11-4
Monitoring for Compliance 11-6
Standard Database Auditing 11-7
Configuring the Audit Trail 11-8
Uniform Audit Trails 11-9
Specifying Audit Options 11-10
Default Auditing 11-11
Enterprise Manager Audit Page 11-12
Using and Maintaining Audit Information 11-13
Value-Based Auditing 11-14
Fine-Grained Auditing 11-16
FGA Policy 11-17
Audited DML Statement: Considerations 11-19
FGA Guidelines 11-20
SYSDBA Auditing 11-21
Maintaining the Audit Trail 11-22

Oracle Audit Vault 11-23
Quiz 11-24
Summary 11-26
Practice 11 Overview: Implementing Oracle Database Security 11-27

12 Database Maintenance

Objectives 12-2
Database Maintenance 12-3
Viewing the Alert History 12-4
Terminology 12-5
Oracle Optimizer: Overview 12-6
Optimizer Statistics 12-7
Using the Manage Optimizer Statistics Page 12-8
Gathering Optimizer Statistics Manually 12-9
Preferences for Gathering Statistics 12-11
Automatic Workload Repository (AWR) 12-13
AWR Infrastructure 12-14
AWR Baselines 12-15
Enterprise Manager and the AWR 12-16
Managing the AWR 12-17
Statistic Levels 12-18
Automatic Database Diagnostic Monitor (ADDM) 12-19
ADDM Findings 12-20
ADDM Recommendations 12-21
Advisory Framework 12-22
Enterprise Manager and Advisors 12-24
DBMS_ADVISOR Package 12-25
Quiz 12-26
Automated Maintenance Tasks 12-27
Automated Maintenance Tasks Configuration 12-29
Server-Generated Alerts 12-30
Setting Thresholds 12-31
Creating and Testing an Alert 12-32
Alerts Notification 12-33
Reacting to Alerts 12-35
Alert Types and Clearing Alerts 12-36
Quiz 12-37
Summary 12-38
Practice 12 Overview: Proactive Maintenance 12-39

13 Performance Management

- Objectives 13-2
- Performance Monitoring 13-3
- Enterprise Manager Performance Page 13-4
- Drilling Down to a Particular Wait Category 13-5
- Performance Page: Throughput 13-6
- Performance Monitoring: Top Sessions 13-7
- Performance Monitoring: Top Services 13-8
- Managing Memory Components 13-9
- Enabling Automatic Memory Management (AMM) 13-10
- Enabling Automatic Shared Memory Management (ASMM) 13-11
- Automatic Shared Memory Advisor 13-12
- Dynamic Performance Statistics 13-13
- Troubleshooting and Tuning Views 13-15
- Invalid and Unusable Objects 13-16
- Quiz 13-18
- Summary 13-20
- Practice 13 Overview: Monitoring and Improving Performance 13-21

14 Backup and Recovery Concepts

- Objectives 14-2
- Part of Your Job 14-3
- Categories of Failure 14-5
- Statement Failure 14-6
- User Process Failure 14-7
- Network Failure 14-8
- User Error 14-9
- Flashback Technology 14-10
- Instance Failure 14-12
- Understanding Instance Recovery: Checkpoint (CKPT) Process 14-13
- Understanding Instance Recovery: Redo Log Files and Log Writer 14-14
- Understanding Instance Recovery 14-15
- Phases of Instance Recovery 14-16
- Tuning Instance Recovery 14-17
- Using the MTTR Advisor 14-18
- Media Failure 14-19
- Configuring for Recoverability 14-20
- Configuring the Fast Recovery Area 14-21
- Multiplexing Control Files 14-22
- Redo Log Files 14-24
- Multiplexing the Redo Log 14-25

Archive Log Files 14-26
Archiver (ARCn) Process 14-27
Archive Log File: Naming and Destinations 14-28
Enabling ARCHIVELOG Mode 14-29
Quiz 14-30
Summary 14-32
Practice 14 Overview: Configuring for Recoverability 14-33

15 Performing Database Backups

Objectives 15-2
Backup Solutions: Overview 15-3
Oracle Secure Backup 15-4
User-Managed Backup 15-5
Terminology 15-6
Recovery Manager (RMAN) 15-8
Configuring Backup Settings 15-9
Scheduling Backups: Strategy 15-11
Scheduling Backups: Options 15-12
Scheduling Backups: Settings 15-13
Scheduling Backups: Schedule 15-14
Scheduling Backups: Review 15-15
Backing Up the Control File to a Trace File 15-16
Managing Backups 15-17
Viewing Backup Reports 15-18
Monitoring the Fast Recovery Area 15-19
Using the RMAN Command Line 15-20
Quiz 15-21
Summary 15-22
Practice 15 Overview: Creating Database Backups 15-23

16 Performing Database Recovery

Objectives 16-2
Opening a Database 16-3
Keeping a Database Open 16-5
Data Recovery Advisor 16-6
Loss of a Control File 16-8
Loss of a Redo Log File 16-9
Loss of a Data File in NOARCHIVELOG Mode 16-11
Loss of a Noncritical Data File in ARCHIVELOG Mode 16-12

Loss of a System-Critical Data File in ARCHIVELOG Mode	16-13
Data Failure: Examples	16-14
Data Recovery Advisor	16-15
Assessing Data Failures	16-16
Data Failures	16-17
Listing Data Failures	16-18
Advising on Repair	16-19
Executing Repairs	16-20
Data Recovery Advisor Views	16-21
Quiz	16-22
Summary	16-24
Practice 16 Overview: Performing Database Recovery	16-25

17 Moving Data

Objectives	17-2
Moving Data: General Architecture	17-3
Oracle Data Pump: Overview	17-4
Oracle Data Pump: Benefits	17-5
Directory Objects for Data Pump	17-7
Creating Directory Objects	17-8
Data Pump Export and Import Clients: Overview	17-9
Data Pump Utility: Interfaces and Modes	17-10
Data Pump Export using Database Control	17-11
Data Pump Export Example: Basic Options	17-12
Data Pump Export Example: Advanced Options	17-13
Data Pump Export Example: Files	17-14
Data Pump Export Example: Schedule	17-16
Data Pump Export Example: Review	17-17
Data Pump Import Example: <code>impdp</code>	17-18
Data Pump Import: Transformations	17-19
Using Enterprise Manager to Monitor Data Pump Jobs	17-20
Migration with Data Pump Legacy Mode	17-21
Data Pump Legacy Mode	17-22
Managing File Locations	17-24
SQL*Loader: Overview	17-25
Loading Data with SQL*Loader	17-27
SQL*Loader Control File	17-28
Loading Methods	17-30
External Tables	17-31
External Table Benefits	17-32
Defining an External Tables with <code>ORACLE_LOADER</code>	17-33

External Table Population with ORACLE_DATAPUMP	17-34
Using External Tables	17-35
Data Dictionary	17-36
Quiz	17-37
Summary	17-39
Practice 17 Overview: Moving Data	17-40

18 Working with Support

Objectives	18-2
Using the Support Workbench	18-3
Viewing Critical Error Alerts in Enterprise Manager	18-4
Viewing Problem Details	18-5
Viewing Incident Details: Dump Files	18-6
Viewing Incident Details: Checker Findings	18-7
Creating a Service Request	18-8
Packaging and Uploading Diagnostic Data to Oracle Support	18-9
Tracking the Service Request and Implementing Repairs	18-10
Closing Incidents and Problems	18-12
Incident Packaging Configuration	18-13
Enterprise Manager Support Workbench for ASM	18-14
Working with Oracle Support	18-15
My Oracle Support Integration	18-16
Using My Oracle Support	18-17
Researching an Issue	18-19
Logging Service Requests	18-21
Managing Patches	18-23
Applying a Patch Release	18-24
Using the Patch Advisor	18-25
Using the Patch Wizard	18-26
Applying a Patch	18-27
Staging a Patch	18-28
Online Patching: Overview	18-29
Installing an Online Patch	18-30
Benefits of Online Patching	18-31
Conventional Patching and Online Patching	18-32
Online Patching Considerations	18-33
Quiz	18-35
Summary	18-36
Practice 18 Overview: Using EM Tools for Alerts and Patches	18-37

Appendix A: Practices and Solutions

Appendix B: Basic Linux and vi Commands

Appendix C: SQL Statement Syntax

Appendix D: Oracle Background Processes

Appendix E: Acronyms and Terms

F Oracle Restart

Objectives F-2
Oracle Restart F-3
Oracle Restart Process Startup F-5
Controlling Oracle Restart F-6
Choosing the Correct SRVCTL Utility F-8
Oracle Restart Configuration F-9
Using the SRVCTL Utility F-10
Obtaining Help for the SRVCTL Utility F-11
Starting Components by Using the SRVCTL Utility F-12
Stopping Components by Using the SRVCTL Utility F-13
Viewing Component Status F-14
Displaying the Oracle Restart Configuration for a Component F-15
Manually Adding Components to the Oracle Restart Configuration F-16
Quiz F-17
Summary F-18
Practice 3-1: Overview F-19

G Continuing Your Education and Further Reading

Where Do You Go from Here? G-2
Continuing Education Resources G-3
Oracle University G-4
Continuing Your Education G-5
Database Specialty Areas G-6
Oracle Real Application Clusters G-7
Oracle Data Guard G-8
Streams Overview G-9
Oracle Technology Network G-11
Security G-12
Oracle by Example G-13
Oracle Magazine G-14

Oracle Applications Community G-15
Technical Support: My Oracle Support G-16
Oracle Database Product Page G-17
Thank You! G-18

14

Backup and Recovery Concepts

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Identify the types of failure that can occur in an Oracle database
- Describe ways to tune instance recovery
- Identify the importance of checkpoints, redo log files, and archive log files
- Configure the fast recovery area
- Configure ARCHIVELOG mode



Copyright © 2009, Oracle. All rights reserved.

Part of Your Job

The database administrator's duties are to:

- Protect the database from failure wherever possible
- Increase the mean time between failures (MTBF)
- Protect critical components by redundancy
- Decrease the mean time to recover (MTTR)
- Minimize the loss of data



Copyright © 2009, Oracle. All rights reserved.

Part of Your Job

The goal of the database administrator (DBA) is to ensure that the database is open and available when users need it. To achieve that goal, the DBA (working with the system administrator):

- Anticipates and works to avoid common causes of failure
- Works to increase the mean time between failures (MTBF) that negatively affect availability
- Ensures that hardware is as reliable as possible, that critical components are protected by redundancy, and that operating system maintenance is performed in a timely manner. The Oracle database provides advanced configuration options to increase MTBF, including:
 - Real Application Clusters (discussed in the *Oracle Database 11g: Real Application Clusters* course)
 - Streams (discussed in the *Oracle Database 11g: Implement Streams* course)
 - Oracle Data Guard (discussed in the *Oracle Database 11g: Data Guard Administration* course)
- Decreases the mean time to recover (MTTR) by practicing recovery procedures in advance and configuring backups so that they are readily available when needed

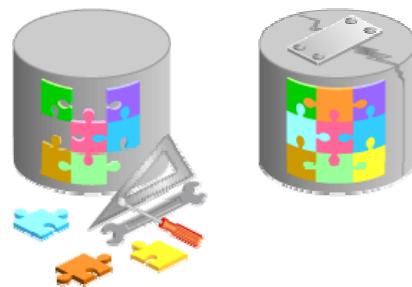
Part of Your Job (continued)

- Minimizes the loss of data. DBAs who follow accepted best practices can configure their databases so that no committed transaction is ever lost. Entities that assist in guaranteeing this include:
 - Archive log files (discussed later in this lesson)
 - Flashback technology
 - Standby databases and Oracle Data Guard (discussed in the *Oracle Database 11g: Data Guard Administration* course)

Categories of Failure

Failures can generally be divided into the following categories:

- Statement failure
- User process failure
- Network failure
- User error
- Instance failure
- Media failure



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Categories of Failure

- **Statement failure:** A single database operation (select, insert, update, or delete) fails.
- **User process failure:** A single database session fails.
- **Network failure:** Connectivity to the database is lost.
- **User error:** A user successfully completes an operation, but the operation (dropping a table or entering bad data) is incorrect.
- **Instance failure:** The database instance shuts down unexpectedly.
- **Media failure:** A loss of any file that is needed for database operation (that is, the files have been deleted or the disk has failed).

Statement Failure

Typical Problems	Possible Solutions
Attempts to enter invalid data into a table	Work with users to validate and correct data.
Attempts to perform operations with insufficient privileges	Provide appropriate object or system privileges.
Attempts to allocate space that fail	<ul style="list-style-type: none">Enable resumable space allocation.Increase owner quota.Add space to tablespace.
Logic errors in applications	Work with developers to correct program errors.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Statement Failure

When a single database operation fails, DBA involvement may be necessary to correct errors with user privileges or database space allocation. DBAs may also need to assist in troubleshooting, even for problems that are not directly in their task area. This can vary greatly from one organization to another. For example, in organizations that use off-the-shelf applications (that is, organizations that have no software developers), the DBA is the only point of contact and must examine logic errors in applications.

To understand logic errors in applications, you should work with developers to understand the scope of the problem. Oracle Database tools may provide assistance by helping to examine audit trails or previous transactions.

Note: In many cases, statement failures are by design and desired. For example, security policies and quota rules are often decided upon in advance. When a user gets an error while trying to exceed his or her limits, it may be desired for the operation to fail and no resolution may be necessary.

User Process Failure

Typical Problems	Possible Solutions
A user performs an abnormal disconnect.	A DBA's action is not usually needed to resolve user process failures. Instance background processes roll back uncommitted changes and release locks.
A user's session is abnormally terminated.	
A user experiences a program error that terminates the session.	Watch for trends. 

ORACLE

Copyright © 2009, Oracle. All rights reserved.

User Process Failure

User processes that abnormally disconnect from the instance may have uncommitted work in progress that needs to be rolled back. The Process Monitor (PMON) background process periodically polls server processes to ensure that their sessions are still connected. If PMON finds a server process whose user is no longer connected, PMON recovers from any ongoing transactions; it also rolls back uncommitted changes and releases any locks that are held by the failed session.

A DBA's intervention should not be required to recover from user process failure, but the administrator must watch for trends. One or two users disconnecting abnormally is not a cause for concern. A small percentage of user process failures may occur from time to time.

But consistent and systemic failures indicate other problems. A large percentage of abnormal disconnects may indicate a need for user training (which includes teaching users to log out rather than just terminate their programs). It may also be indicative of network or application problems.

Network Failure

Typical Problems	Possible Solutions
Listener fails.	Configure a backup listener and connect-time failover.
Network Interface Card (NIC) fails.	Configure multiple network cards.
Network connection fails.	Configure a backup network connection.

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Network Failure

The best solution to network failure is to provide redundant paths for network connections. Backup listeners, network connections, and network interface cards reduce the chance that network failures will affect system availability.

User Error

Typical Causes	Possible Solutions
User inadvertently deletes or modifies data.	Roll back transaction and dependent transactions or rewind table.
User drops a table.	Recover table from recycle bin.



ORACLE®

Copyright © 2009, Oracle. All rights reserved.

User Error

Users may inadvertently delete or modify data. If they have not yet committed or exited their program, they can simply roll back.

You can use Oracle LogMiner to query your online redo logs and archived redo logs through an Enterprise Manager or SQL interface. Transaction data may persist in online redo logs longer than it persists in undo segments; if you have configured archiving of redo information, redo persists until you delete the archived files. Oracle LogMiner is discussed in the *Oracle Database Utilities* reference.

Users who drop a table can recover it from the recycle bin by flashing back the table to before the drop. Flashback technologies are discussed in detail in the *Oracle Database 11g: Administration Workshop II* course.

If the recycle bin has already been purged, or if the user dropped the table with the PURGE option, the dropped table can still be recovered by using point-in-time recovery (PITR) if the database has been properly configured. PITR is discussed in the *Oracle Database 11g: Administration Workshop II* course and in the *Oracle Database Backup and Recovery User's Guide*.

Flashback Technology

Using Flashback technology:

- Viewing past states of data
- Winding data back and forth in time
- Assisting users in error analysis and recovery



For error analysis:

Oracle Flashback Query

Oracle Flashback Versions Query

Oracle Flashback Transaction Query

For error recovery:

Oracle Flashback Transaction Backout

Oracle Flashback Table

Oracle Flashback Drop

Oracle Flashback Database

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Flashback Technology

The Oracle database provides Oracle Flashback technology: a group of features that support viewing past states of data—and winding data back and forth in time—without requiring restoring the database from backup. With this technology, you help users analyze and recover from errors. For users who have committed erroneous changes, use the following to analyze the errors:

- **Flashback Query:** View committed data as it existed at some point in the past. The SELECT command with the AS OF clause references a time in the past through a time stamp or SCN.
- **Flashback Version Query:** View committed historical data for a specific time interval. Use the VERSIONS BETWEEN clause of the SELECT command (for performance reasons with existing indexes).
- **Flashback Transaction Query:** View all database changes made at the transaction level

Possible solutions to recover from user error:

- **Flashback Transaction Backout:** Rolls back a specific transaction and dependent transactions
- **Flashback Table:** Rewinds one or more tables to their contents at a previous time without affecting other database objects

Flashback Technology (continued)

- **Flashback Drop:** Reverses the effects of dropping a table by returning the dropped table from the recycle bin to the database along with dependent objects such as indexes and triggers
- **Flashback Database:** Returns the database to a past time or system change number (SCN)

Instance Failure

Typical Causes	Possible Solutions
Power outage	Restart the instance by using the STARTUP command. Recovering from instance failure is automatic, including rolling forward changes in the redo logs and then rolling back any uncommitted transactions.
Hardware failure	Investigate the causes of failure by using the alert log, trace files, and Enterprise Manager.
Failure of one of the critical background processes	
Emergency shutdown procedures	

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Instance Failure

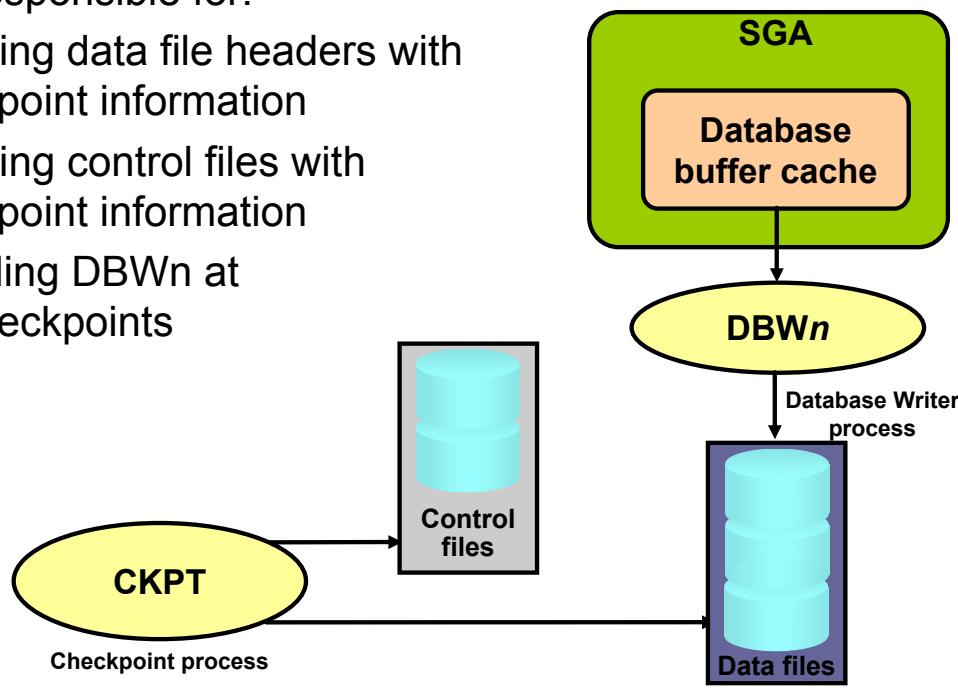
Instance failure occurs when the database instance is shut down before synchronizing all database files. An instance failure can occur because of hardware or software failure or through the use of the emergency SHUTDOWN ABORT and STARTUP FORCE shutdown commands.

Administrator involvement in recovering from instance failure rarely required if Oracle Restart enabled and monitoring your database. Oracle Restart attempts to restart your database instance as soon as it fails. If manual intervention is required then there may be a more serious problem that prevents the instance from restarting, such as a memory CPU failure.

Understanding Instance Recovery: Checkpoint (CKPT) Process

CKPT is responsible for:

- Updating data file headers with checkpoint information
- Updating control files with checkpoint information
- Signaling DBWn at full checkpoints



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Understanding Instance Recovery: Checkpoint (CKPT) Process

To understand instance recovery, you need to understand the functioning of certain background processes.

Every three seconds (or more frequently), the CKPT process stores data in the control file to document the modified data blocks that DBW n has written from the SGA to disk. This is called an “incremental checkpoint.” The purpose of a checkpoint is to identify that place in the online redo log file where instance recovery is to begin (which is called the “checkpoint position”).

In the event of a log switch, the CKPT process also writes this checkpoint information to the headers of data files.

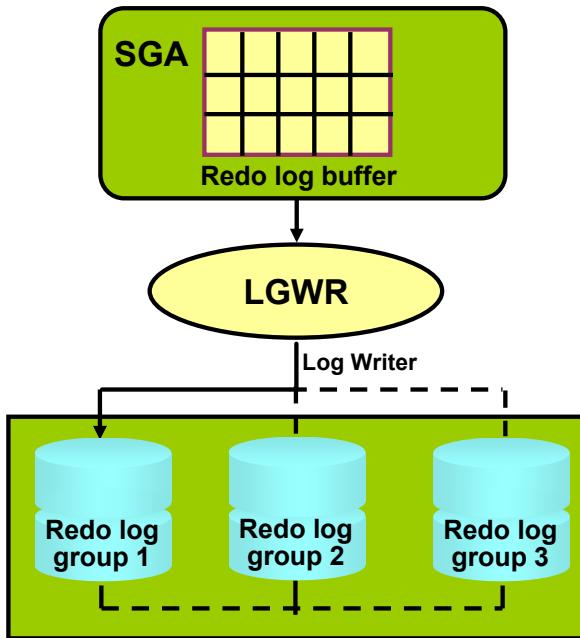
Checkpoints exist for the following reasons:

- To ensure that modified data blocks in memory are written to the disk regularly so that data is not lost in case of a system or database failure
- To reduce the time required for instance recovery (Only the online redo log file entries following the last checkpoint need to be processed for recovery.)
- To ensure that all committed data has been written to data files during shutdown

The checkpoint information written by the CKPT process includes checkpoint position, system change number (SCN), location in the online redo log file to begin recovery, information about logs, and so on.

Note: The CKPT process does not write data blocks to the disk or redo blocks to the online redo log files.

Understanding Instance Recovery: Redo Log Files and Log Writer



Redo log files:

- Record changes to the database
- Should be multiplexed to protect against loss

Log Writer writes:

- At commit
- When one-third full
- Every three seconds
- Before DBW n writes
- Before clean shutdowns

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Understanding Instance Recovery: Redo Log Files and Log Writer

Redo log files record changes to the database as a result of transactions and internal Oracle server actions. (A transaction is a logical unit of work consisting of one or more SQL statements run by a user.) Redo log files protect the database from the loss of integrity because of system failures caused by power outages, disk failures, and so on. Redo log files should be multiplexed to ensure that the information stored in them is not lost in the event of a disk failure.

The redo log consists of groups of redo log files. A group consists of a redo log file and its multiplexed copies. Each identical copy is said to be a member of that group, and each group is identified by a number. The Log Writer (LGWR) process writes redo records from the redo log buffer to all members of a redo log group until the files are filled or a log switch operation is requested.

It then switches and writes to the files in the next group. Redo log groups are used in a circular fashion.

Best practice tip: If possible, multiplexed redo log files should reside on different disks.

Understanding Instance Recovery

Automatic instance or crash recovery:

- Is caused by attempts to open a database whose files are not synchronized on shutdown
- Uses information stored in redo log groups to synchronize files
- Involves two distinct operations:
 - Rolling forward: Redo log changes (both committed and uncommitted) are applied to data files.
 - Rolling back: Changes that are made but not committed are returned to their original state.

ORACLE

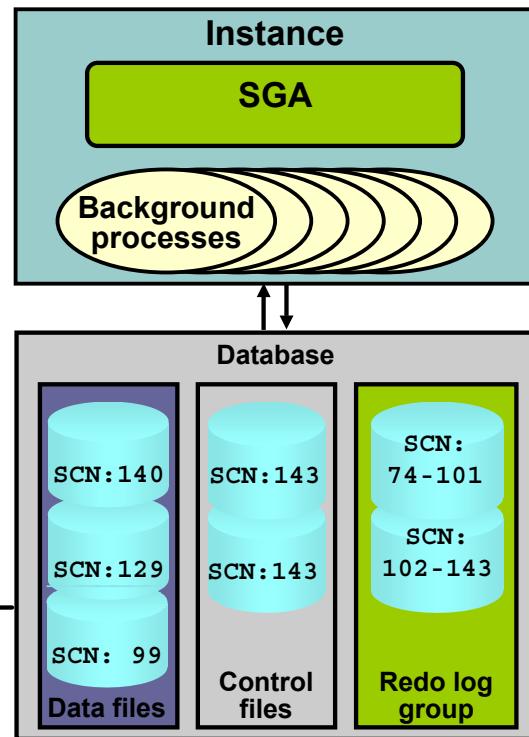
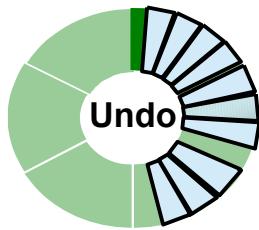
Copyright © 2009, Oracle. All rights reserved.

Instance Recovery

The Oracle database automatically recovers from instance failure. All that needs to happen is for the instance to be started normally. If Oracle Restart is enabled and configured to monitor this database then this happens automatically. The instance mounts the control files and then attempts to open the data files. When it discovers that the data files have not been synchronized during shutdown, the instance uses information contained in the redo log groups to roll the data files forward to the time of shutdown. Then the database is opened and any uncommitted transactions are rolled back.

Phases of Instance Recovery

1. Startup instance (data files are out of sync)
2. Roll forward (redo)
3. Committed and uncommitted data in files
4. Database opened
5. Roll back (undo)
6. Committed data in files



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Phases of Instance Recovery

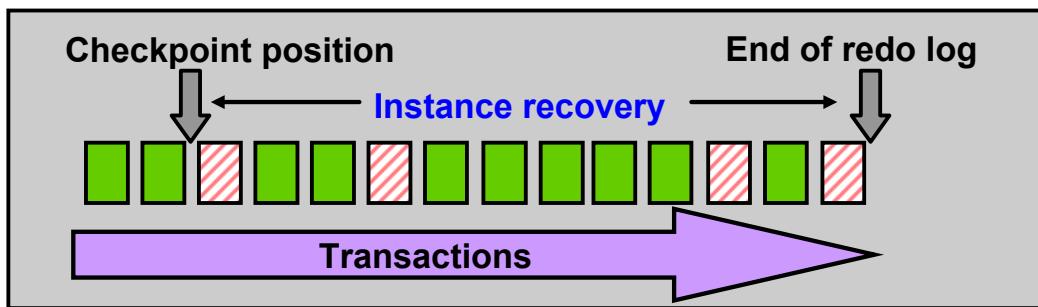
For an instance to open a data file, the system change number (SCN) contained in the data file's header must match the current SCN that is stored in the database's control files.

If the numbers do not match, the instance applies redo data from the online redo logs, sequentially "redoing" transactions until the data files are up-to-date. After all data files have been synchronized with the control files, the database is opened and users can log in.

When redo logs are applied, *all* transactions are applied to bring the database up to the state as of the time of failure. This usually includes transactions that are in progress but have not yet been committed. After the database has been opened, those uncommitted transactions are rolled back. At the end of the rollback phase of instance recovery, the data files contain only committed data.

Tuning Instance Recovery

- During instance recovery, the transactions between the checkpoint position and the end of redo log must be applied to data files.
- You tune instance recovery by controlling the difference between the checkpoint position and the end of redo log.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Tuning Instance Recovery

Transaction information is recorded in the redo log groups before the instance returns `commit complete` for a transaction. The information in the redo log groups guarantees that the transaction can be recovered in case of a failure. The transaction information also needs to be written to the data file. The data file write usually happens at some time after the information is recorded in redo log groups because the data file write process is much slower than the redo writes. (Random writes for data files are slower than serial writes for redo log files.)

Every three seconds, the checkpoint process records information in the control file about the checkpoint position in the redo log. Therefore, the Oracle database knows that all redo log entries recorded before this point are not necessary for database recovery. In the graphic in the slide, the striped blocks have not yet been written to the disk.

The time required for instance recovery is the time required to bring data files from their last checkpoint to the latest SCN recorded in the control file. The administrator controls that time by setting an MTTR target (in seconds) and through the sizing of redo log groups. For example, for two redo groups, the distance between the checkpoint position and the end of the redo log group cannot be more than 90% of the smallest redo log group.

Using the MTTR Advisor

- Specify the desired time in seconds or minutes.
- The default value is 0 (disabled).
- The maximum value is 3,600 seconds (one hour).

Database Instance: orcl.oracle.com

Home Performance Availability Server Schema Data Mo

High Availability Console

Backup/Recovery

Setup Manage

Backup Settings Schedule Backup

Recovery Settings Manage Current Backups

Recovery Catalog Settings Backup Reports

Recovery Settings

Instance Recovery

The fast-start checkpointing feature is enabled by specifying a non-zero desired mean-time to recover (MTTR) value, which will be used to set the FAST_START_MTTR_TARGET initialization parameter. This parameter controls the amount of time the database takes to perform crash recovery for a single instance. When fast-start checkpointing is enabled, Oracle automatically maintains the speed of checkpointing so that the requested MTTR is achieved. Setting the value to 0 will disable this functionality.

Current Estimated Mean Time To Recover (seconds) 15

Desired Mean Time To Recover 0 Minutes

ORACLE

Using the MTTR Advisor

For assistance in setting the MTTR target, select either of the following:

- Enterprise Manager > Advisor Central (in the Related Links section) > MTTR Advisor
- Enterprise Manager > Availability > Recovery Settings

The FAST_START_MTTR_TARGET initialization parameter simplifies the configuration of recovery time from instance or system failure. The MTTR Advisor converts the FAST_START_MTTR_TARGET value into several parameters to enable instance recovery in the desired time (or as close to it as possible). Please note, explicitly setting the FAST_START_MTTR_TARGET parameter to 0 disables the MTTR advisor.

The FAST_START_MTTR_TARGET parameter must be set to a value that supports the service level agreement for your system. A small value for the MTTR target increases I/O overhead because of additional data file writes (affecting the performance). However, if you set the MTTR target too large, the instance takes longer to recover after a crash.

Media Failure

Typical Causes	Possible Solutions
Failure of disk drive	<ol style="list-style-type: none">1. Restore the affected file from backup.
Failure of disk controller	<ol style="list-style-type: none">2. Inform the database about a new file location (if necessary).
Deletion or corruption of a file needed for database operation	<ol style="list-style-type: none">3. Recover the file by applying redo information (if necessary).



Copyright © 2009, Oracle. All rights reserved.

Media Failure

Oracle Corporation defines *media failure* as any failure that results in the loss or corruption of one or more database files (data, control, or redo log file).

Recovering from media failure requires that you restore and recover the missing files. To ensure that your database can be recovered from media failure, follow the best practices outlined in the next few pages.

Configuring for Recoverability

To configure your database for maximum recoverability, you must:

- Schedule regular backups
- Multiplex control files
- Multiplex redo log groups
- Retain archived copies of redo logs



The screenshot shows the Oracle Enterprise Manager interface. The top navigation bar includes links for Home, Performance, Availability (which is selected and highlighted in blue), Server, Schema, Data Movement, and Software and Support. The main content area has a left sidebar with 'Backup/Recovery' and a right sidebar for 'Oracle Secure Backup'. The left sidebar contains 'Setup' with 'Backup Settings' and 'Recovery Settings' (which is highlighted with a red box), and 'Recovery Catalog Settings'. The right sidebar contains 'Manage' with 'Schedule Backup', 'Manage Current Backups', 'Backup Reports', 'Manage Restore Points', 'Perform Recovery', and 'View and Manage Transactions'. The bottom of the interface features a red footer bar with the 'ORACLE' logo and the text 'Copyright © 2009, Oracle. All rights reserved.'

Configuring for Recoverability

To provide the best protection for your data, you must:

- **Schedule regular backups**
Most media failures require that you restore the lost or damaged file from backup.
- **Multiplex control files**
All control files associated with a database are identical. Recovering from the loss of a single control file is not difficult; recovering from the loss of *all* control files is much more challenging. Guard against losing all control files by having at least two copies.
- **Multiplex redo log groups**
To recover from instance or media failure, redo log information is used to roll data files forward to the last committed transaction. If your redo log groups rely on a single redo log file, the loss of that file means that data is likely to be lost. Ensure that there are at least two copies of each redo log group; if possible, each copy should be under different disk controllers.
- **Retain archived copies of redo logs**
If a file is lost and restored from backup, the instance must apply redo information to bring that file up to the latest SCN contained in the control file. With the default setting, the database can overwrite redo information after it has been written to the data files. Your database can be configured to retain redo information in archived copies of the redo logs. This is known as placing the database in ARCHIVELOG mode.

You can perform configuration tasks in Enterprise Manager or with the command line.

Configuring the Fast Recovery Area

Fast recovery area:

- Strongly recommended for simplified backup storage management
- Storage space (separate from working database files)
- Location specified by the `DB_RECOVERY_FILE_DEST` parameter
- Size specified by the `DB_RECOVERY_FILE_DEST_SIZE` parameter
- Large enough for backups, archived logs, flashback logs, multiplexed control files, and multiplexed redo logs
- Automatically managed according to your retention policy

Configuring the fast recovery area means determining location, size, and retention policy.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Configuring the Fast Recovery Area

The fast recovery area is a space that is set aside on the disk to contain archived logs, backups, flashback logs, multiplexed control files, and multiplexed redo logs. A fast recovery area simplifies backup storage management and is strongly recommended. You should place the fast recovery area on storage space that is separate from the location of your database data files and primary online log files and control file.

The amount of disk space to allocate for the fast recovery area depends on the size and activity levels of your database. As a general rule, the larger the fast recovery area, the more useful it is. Ideally, the fast recovery area should be large enough for copies of your data and control files and for flashback, online redo, and archived logs needed to recover the database with the backups kept based on the retention policy. (In short, the fast recovery area should be at least twice the size of the database so that it can hold one backup and several archived logs.)

Space management in the fast recovery area is governed by a backup retention policy. A retention policy determines when files are obsolete, which means that they are no longer needed to meet your data recovery objectives. The Oracle database automatically manages this storage by deleting files that are no longer needed.

Multiplexing Control Files

To protect against database failure, your database should have multiple copies of the control file.

	ASM Storage	File System Storage
Best Practice	One copy on each disk group (such as +DATA and +FRA)	At least two copies, each on separate disk (at least one on separate disk controller)
Steps to create additional control files	No additional control file copies required.	<ol style="list-style-type: none">1. Alter the SPFILE with the ALTER SYSTEM SET control_files command.2. Shut down the database.3. Copy control file to a new location.4. Open the database and verify the addition of the new control file.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Multiplexing Control Files

A control file is a small binary file that describes the structure of the database. It must be available for writing by the Oracle server whenever the database is mounted or opened. Without this file, the database cannot be mounted, and recovery or re-creation of the control file is required. Your database should have a minimum of two control files on different storage devices to minimize the impact of a loss of one control file.

The loss of a single control file causes the instance to fail because all control files must be available at all times. However, recovery can be a simple matter of copying one of the other control files. The loss of all control files is slightly more difficult to recover from but is not usually catastrophic.

Multiplexing Control Files (continued)

Adding a Control File

If using ASM as your storage technique, then as long as you have two control files, one in each disk group (such as +DATA and +FRA), then you should not require further multiplexing. In a database using OMF (such as a database using ASM storage), all additional control files must be created as part of a recovery process using RMAN (or through Enterprise Manager). In a database using regular file system storage, adding a control file is a manual operation:

1. Alter the SPFILE with the following command:

```
ALTER SYSTEM SET control_files =
  '/u01/app/oracle/oradata/orcl/control01.ctl' ,
  '/u02/app/oracle/oradata/orcl/control02.ctl' ,
  '/u03/app/oracle/oradata/orcl/control03.ctl' SCOPE=SPFILE;
```

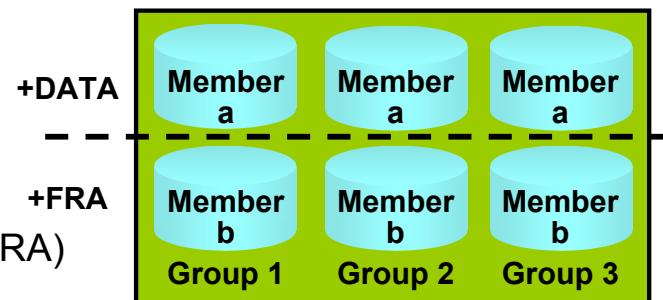
2. Shut down the database.
3. Use the operating system to copy an existing control file to the location you select for your new file.
4. Open the database.

Note: More information on using RMAN is covered in the course titled *Oracle Database 11g: Administration Workshop II* and online product documentation.

Redo Log Files

Multiplex redo log groups to protect against media failure and loss of data. This increases database I/O. It is suggested that redo log groups have:

- At least two members (files) per group
- Each member:
 - On a separate disk or controller if using file system storage
 - In a separate disk group (such as +DATA and +FRA) if using ASM



Note: Multiplexing redo logs may impact overall database performance.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Redo Log Files

Redo log groups are made up of one or more redo log files. Each log file in a group is a duplicate of the others. Oracle Corporation recommends that redo log groups have at least two files per group. If using file system storage, then each member should be distributed on separate disks or controllers so that no single equipment failure destroys an entire log group. If using ASM storage, then each member should be in a separate disk group, such as +DATA and +FRA.

The loss of an entire current log group is one of the most serious media failures because it can result in loss of data. The loss of a single member of a multiple-member log group is trivial and does not affect database operation (other than causing an alert to be published in the alert log). Recovery from the loss of an entire log group requires advanced recovery techniques and is discussed in the course titled *Oracle Database 11g: Administration Workshop II*.

Remember that multiplexing redo logs may heavily influence database performance because a commit cannot complete until the transaction information has been written to the logs. You must place your redo log files on your fastest disks served by your fastest controllers. If possible, do not place any other database files on the same disks as your redo log files (unless you are using Automatic Storage Management [ASM]). Because only one group is written to at a given time, there is no performance impact in having members from several groups on the same disk.

Multiplexing the Redo Log

The screenshot shows the Oracle Enterprise Manager interface for managing redo log groups. A yellow callout box on the left states: "If Storage Type is File System, then you are prompted to enter a File Name and File Directory." The main window displays the "Edit Redo Log Group: 1: Add Redo Log Member" dialog and the "Edit Redo Log Group" page.

Edit Redo Log Group: 1: Add Redo Log Member

- Storage Type: Automatic Storage Management
- * DiskGroup: DATA
- Template: <Default>
- Alias Directory: (empty)
- Alias Name: (empty)
- Reuse File

Edit Redo Log Group

Actions: Clear logfile

Group # 1

File size 51200 KB

Status ACTIVE

Redo Log Members

Object Name: (empty)

By default, the search returns all uppercase-sensitive match, double quote the

Selection Mode: Single

Select File Name

File Name	File Directory
group_1.261.689304441	+DATA/orcl/onlinelog/
group_1.257.689304447	+FRA/orcl/onlinelog/

Add (highlighted with a red box and arrow)

Edit (highlighted with a red box and arrow)

Select Group

Select	Group	Status	# of Members	Archived	Size (KB)	Sequence	First Change#
<input checked="" type="radio"/>	1	Active	2	No	51200	7	834285
<input type="radio"/>	2	Active	2	No	51200	8	849739
<input type="radio"/>	3	Current	2	No	51200	9	849745

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Multiplexing the Redo Log

You can multiplex your redo log by adding a member to an existing log group. To add a member to a redo log group (with open database and no impact on user performance), perform the following steps:

1. Select Enterprise Manager > Server > Redo Log Groups.
2. Select a group and click the Edit button, or click the group number link. The Edit Redo Log Group page appears.
3. In the Redo Log Members region, click Add. The Add Redo Log Member page appears.
4. Choose the appropriate Storage Type, and enter the required information. For ASM, choose the disk group and, if desired, specify template and alias information. For File System storage, enter the file name and the file directory. Click Continue.

Repeat these steps for every existing group that you want to multiplex. An example showing the SQL syntax of adding a redo log member to redo log group 1 (using ASM) is shown here:

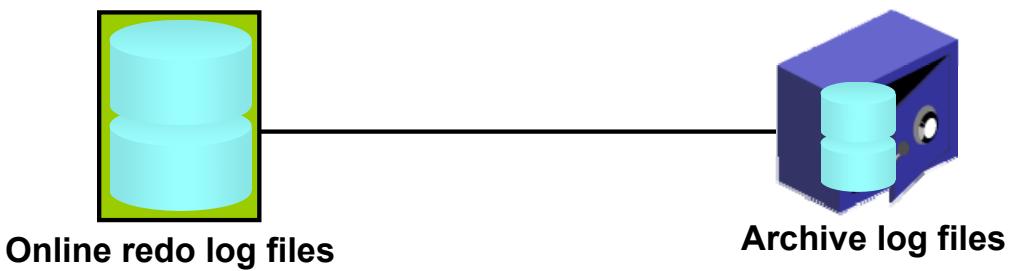
```
SQL> ALTER DATABASE ADD LOGFILE MEMBER '+DATA' TO GROUP 1;
```

When you add the redo log member to a group, the member's status is marked as INVALID (as can be seen in the V\$LOGFILE view). This is the expected state because the new member of the group has not yet been written to. When a log switch occurs and the group containing the new member becomes CURRENT, the member's status changes to null.

Archive Log Files

To preserve redo information, create archived copies of redo log files by performing the following steps.

1. Specify archive log file-naming convention.
2. Specify one or more archive log file locations.
3. **Switch the database to ARCHIVELOG mode.**



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Archive Log Files

The instance treats the online redo log groups as a circular buffer in which to store transaction information, filling one group and then moving on to the next. After all groups have been written to, the instance begins overwriting information in the first log group.

To configure your database for maximum recoverability, you must instruct the database to make a copy of the online redo log group before allowing it to be overwritten. These copies are known as *archived logs*.

To facilitate the creation of archive log files:

1. Specify a naming convention for your archive logs.
2. Specify a destination or destinations for storing your archive logs. One of the destinations is probably your fast recovery area.
3. Place the database in ARCHIVELOG mode.

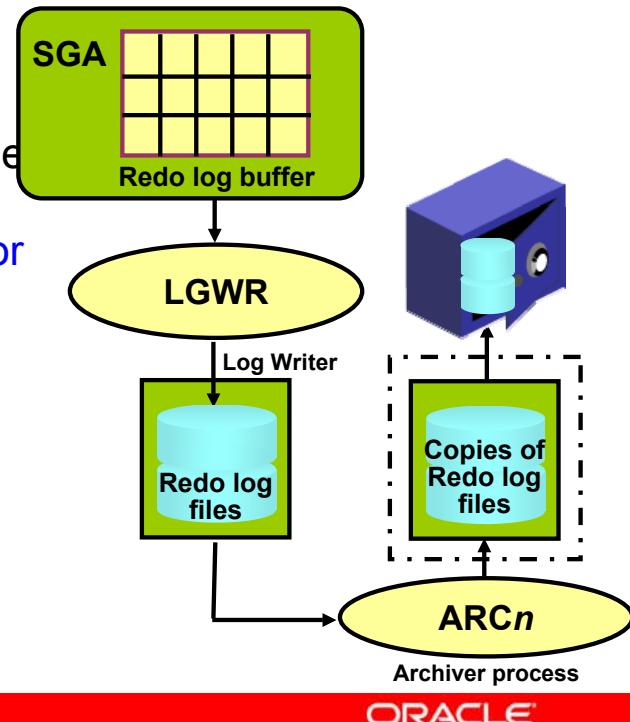
Note: Steps 1 and 2 are not necessary if you are using a fast recovery area.

The destination should exist before placing the database in ARCHIVELOG mode. When a directory is specified as a destination, there should be a slash at the end of the directory name.

Archiver (ARCn) Process

Archiver (ARCn):

- Is an optional background process
- Automatically archives online redo log files **when ARCHIVELOG mode is set for the database**
- Preserves the record of all changes made to the database



Copyright © 2009, Oracle. All rights reserved.

The Archiver (ARCn) Process

ARCn is an optional background process. However, it is crucial to the recovery of a database after the loss of a disk. When an online redo log group gets filled, the Oracle instance begins writing to the next online redo log group. The process of switching from one online redo log group to another is called a *log switch*. The ARCn process initiates archiving of the filled log group at every log switch. It automatically archives the online redo log group before the log group can be reused so that all the changes made to the database are preserved. This enables recovery of the database to the point of failure even if a disk drive is damaged.

One of the important decisions that a DBA must make is whether to configure the database to operate in ARCHIVELOG mode or in NOARCHIVELOG mode.

- In NOARCHIVELOG mode, the online redo log files are overwritten each time a log switch occurs.
- In ARCHIVELOG mode, inactive groups of filled online redo log files must be archived before they can be used again.

Note

- ARCHIVELOG mode is essential for most backup strategies (and is very easy to configure).
- If the Archive log file destination fills up or cannot be written, the database will eventually come to a halt. Remove archive files from the archive log file destination and the database will resume operations.

Archive Log File: Naming and Destinations

Specify naming and archive destination information on the Recovery Settings page. If using file system storage then it is recommended that you add multiple locations across different disks.

Media Recovery

The database is currently in ARCHIVELOG mode. In ARCHIVELOG mode, hot backups and recovery to the latest time are possible, but you must provide space for archived redo log files. If you change the database to ARCHIVELOG mode, you should perform a backup immediately. In NOARCHIVELOG mode, only cold backups are possible and data may be lost in the event of database corruption.

ARCHIVELOG Mode*

Log Archive Filename Format* `%t_%s_%r.dbf`

Number	Archived Redo Log Destination	Status	Type
1	USE_DB_RECOVERY_FILE_DEST	VALID	Local

TIP It is recommended that archived redo log files be written to multiple locations spread across the different disks.
 TIP You can specify up to 10 archived redo log destinations.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Archive Log File: Naming and Destinations

To configure archive log file names and destinations, select Enterprise Manager > Availability > Configure Recovery Settings.

Each archive log file must have a unique name to avoid overwriting older log files. Specify the naming format as shown in the slide. To help create unique file names, Oracle Database 11g allows several wildcard characters in the name format:

- **%s:** Includes the log sequence number as part of the file name
- **%t:** Includes the thread number as part of the file name
- **%r:** Includes the resetlogs ID to ensure that the archive log file name remains unique (even after certain advanced recovery techniques that reset log sequence numbers)
- **%d:** Includes the database ID as part of the file name

The format should include %s, %t, and %r as best practice (%d can also be included if multiple databases share the same archive log destination).

By default if the fast recovery area is enabled, then USE_DB_RECOVERY_FILE_DEST is specified as an archive log file destination. Archive log files can be written to as many as ten different destinations. Destinations may be local (a directory) or remote (an Oracle Net alias for a standby database). Click Add Another Row to add further destinations. To change recovery settings, you must be connected as SYSDBA or SYSOPER.

Note: If you do not want archives sent to this location, delete USE_DB_RECOVERY_FILE_DEST.

Enabling ARCHIVELOG Mode

To place the database in ARCHIVELOG mode, perform the following steps in Enterprise Manager:

1. Select the ARCHIVELOG Mode check box and click Apply.
The database can be set to ARCHIVELOG mode only from the MOUNT state.
2. Restart the database (with SYSDBA privileges).
3. (Optional) View the archive status.
4. Back up your database.

Note: Databases in ARCHIVELOG mode have access to the full range of backup and recovery options.

```
sqlplus / as sysdba
shutdown immediate
startup mount
alter database archivelog;
alter database open;
archive log list
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Enabling ARCHIVELOG Mode

1. In Enterprise Manager, select Availability > Configure Recovery Settings > ARCHIVELOG Mode. The equivalent SQL command is:
SQL> ALTER DATABASE ARCHIVELOG;
This command can be issued only while the database is in the MOUNT state. The instance must therefore be restarted to complete this last step.
2. In Enterprise Manager, you are prompted for operating system and database credentials during the restart of the database. The database credentials *must* be for a user with the SYSDBA privileges.
3. After the instance is restarted, the changes that you have made to the archive processes, log format, and log destinations are in effect. In SQL*Plus, you can see them with the ARCHIVE LOG LIST command.
4. Back up your database after switching to ARCHIVELOG mode because *your database is only recoverable from the last backup taken in that mode*.

With the database in NOARCHIVELOG mode (the default), recovery is possible only until the time of the last backup. All transactions made after that backup are lost.

In ARCHIVELOG mode, recovery is possible until the time of the last commit. Most production databases are run in ARCHIVELOG mode.

Quiz

Statement failure is never by design and always requires the DBA to address the issue.

1. True
2. False



Copyright © 2009, Oracle. All rights reserved.

Answer: 2

Quiz

Which parameters configure the fast recovery area?

1. FLASH_RECOVERY_AREA_SIZE
2. DB_RECOVERY_FILE_DEST
3. FLASH_RECOVERY_AREA_LOC
4. DB_RECOVERY_FILE_DEST_SIZE

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answers: 2, 4

Summary

In this lesson, you should have learned how to:

- Identify the types of failure that can occur in an Oracle database
- Describe ways to tune instance recovery
- Identify the importance of checkpoints, redo log files, and archive log files
- Configure the fast recovery area
- Configure ARCHIVELOG mode



Copyright © 2009, Oracle. All rights reserved.

Practice 14 Overview: Configuring for Recoverability

This practice covers the following topics:

- Verifying control files
- Configuring a default fast recovery area
- Multiplexing redo log groups
- Placing your database in ARCHIVELOG mode
- Ensuring that redundant archive logs are created



Copyright © 2009, Oracle. All rights reserved.

15

Performing Database Backups

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Create consistent database backups
- Back up your database without shutting it down
- Create incremental backups
- Automate database backups
- Manage backups and view backup reports
- Monitor the fast recovery area

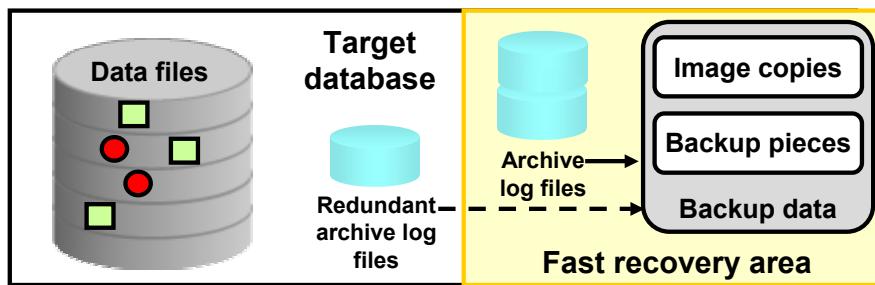


Copyright © 2009, Oracle. All rights reserved.

Backup Solutions: Overview

Backups can be performed by using:

- Recovery Manager
- Oracle Secure Backup
- User-managed backup



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Backup Solutions: Overview

As you will see in the remainder of this lesson, Recovery Manager (RMAN) is the recommended method of backing up your Oracle database.

Oracle Secure Backup complements existing functionality by adding backup to tape and backup of operating system files.

User-managed backups are based on scripts that a DBA must write. This option is being phased out because it is more labor intensive.

Oracle Secure Backup

- Oracle Secure Backup and RMAN provide an end-to-end backup solution for Oracle environments:
 - Centralized tape backup management for file system data and the Oracle database
 - Most well-integrated media management layer for RMAN backups
 - Backup of any data anywhere on the network
- A single technical support resource for the entire backup solution expedites problem resolution.
- This ensures reliable data protection at lower cost and complexity.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Secure Backup

Oracle's current backup and recovery product for the database is Recovery Manager. Oracle Secure Backup complements existing functionality in the following ways:

- **Complete backup solution:** Oracle Secure Backup provides data protection for the database and nondatabase data to protect the entire Oracle environment.
- **Media management:** Oracle Secure Backup provides the media management layer for RMAN database backups to tape. Before Oracle Secure Backup, customers had to purchase expensive third-party media management products offering integration with RMAN tape backups.
- **Backup anywhere on the network:** Oracle Secure Backup backs up data from multiple network-attached computer systems to tertiary storage resources on the network. Oracle Secure Backup supports diverse configurations of servers, clients, Network Attached Storage (NAS) servers, and tertiary storage devices and protects network storage environments.

The combination of RMAN and Oracle Secure Backup provides an end-to-end backup solution that is entirely within the Oracle product stack. This solution makes better customer support possible because Oracle Corporation is responsible for the entire backup solution.

User-Managed Backup

A user-managed scenario:

- Is a manual process of tracking backup needs and status
- Typically uses your own written scripts
- Requires that database files be put in the correct mode for backup
- Relies on operating system commands to make backups of files



Copyright © 2009, Oracle. All rights reserved.

User-Managed Backup

A user-managed backup can be performed interactively. However, most often it entails the writing of scripts to perform the backup. There are several scenarios that can be run, and scripts must be written to handle them.

Some of the actions that scripts must take:

- Querying V\$DATAFILE to determine the data files that need to be backed up and their current state
- Querying V\$LOGFILE to identify the online redo log files
- Querying V\$CONTROLFILE to identify the control file to back up
- Placing each tablespace in online backup mode
- Querying V\$BACKUP to see what data files are part of a tablespace that has been placed in online backup mode
- Issuing operating system copy commands to copy the data files to the backup location
- Bringing each tablespace out of online backup mode

Terminology

- Backup strategy may include:
 - Entire database (whole)
 - Portion of the database (partial)
- Backup type may indicate inclusion of:
 - All data blocks within your chosen files (full)
 - Only information that has changed since a previous backup (incremental)
 - Cumulative (changes since last level 0)
 - Differential (changes since last incremental)
- Backup mode may be:
 - Offline (consistent, cold)
 - Online (inconsistent, hot)



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Terminology

Whole database backup: Includes all data files and at least one control file (Remember that all control files in a database are identical.)

Partial database backup: May include zero or more tablespaces and zero or more data files; may or may not include a control file

Full backup: Makes a copy of each data block that contains data and that is within the files being backed up

Incremental backup: Makes a copy of all data blocks that have changed since a previous backup. The Oracle database supports two levels of incremental backup (0 and 1). A level 1 incremental backup can be one of two types: *cumulative* or *differential*. A cumulative backup backs up all changes since the last level 0 backup. A differential backup backs up all changes since the last incremental backup (which could be either a level 0 or level 1 backup). Change Tracking with RMAN supports incremental backups.

Offline backups (also known as “cold” or *consistent* backup): Are taken while the database is not open. They are consistent because, at the time of the backup, the system change number (SCN) in data file headers matches the SCN in the control files.

Online backups (also known as “hot” or *inconsistent* backup): Are taken while the database is open. They are inconsistent because, with the database open, there is no guarantee that the data files are synchronized with the control files.

Terminology

Backups may be stored as:

- Image copies
- Backup sets

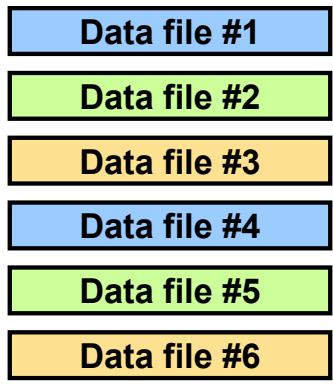
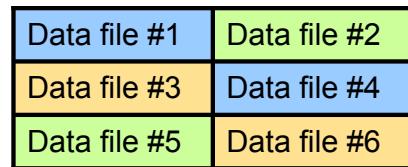


Image copies
(Duplicate data and log files in OS format)



Backup set
(Binary, compressed files in
Oracle proprietary format)

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Terminology (continued)

Image copies: Are duplicates of data or archived log files (similar to simply copying the files by using operating system commands)

Backup sets: Are collections of one or more binary files that contain one or more data files, control files, server parameter files, or archived log files. With backup sets, empty data blocks are not stored, thereby causing backup sets to use less space on the disk or tape. Backup sets can be compressed to further reduce the space requirements of the backup.

Image copies must be backed up to the disk. Backup sets can be sent to the disk or directly to the tape.

The advantage of creating a backup as an image copy is improved granularity of the restore operation. With an image copy, only the file or files need to be retrieved from your backup location. With backup sets, the entire backup set must be retrieved from your backup location before you extract the file or files that are needed.

The advantage of creating backups as backup sets is better space usage. In most databases, 20% or more of the data blocks are empty blocks. Image copies back up every data block, even if the data block is empty. Backup sets significantly reduce the space required by the backup. In most systems, the advantages of backup sets outweigh the advantages of image copies.

Recovery Manager (RMAN)

- Powerful control and scripting language
- Integrated with Enterprise Manager
- Published API that enables interface with most popular backup software
- Backing up data, control, archived log, and server parameter files
- Backing up files to the disk or tape



The screenshot shows a navigation bar with tabs: Home, Performance, Availability, Server (which is selected), Schema, Data Movement, and Software and Support. Under the Server tab, there is a sub-menu titled 'Backup/Recovery' with the following options: Setup (Backup Settings, Recovery Settings, Recovery Catalog Settings), Manage (Schedule Backup, Manage Current Backups, Backup Reports, Manage Restore Points, Perform Recovery, View and Manage Transactions), and Oracle Secure Backup (Assign and Manage).

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Recovery Manager (RMAN)

RMAN is the component of the Oracle database that is used to perform backup and recovery operations. It can make consistent and inconsistent backups, perform incremental and full backups, and back up either the whole database or a portion of it.

RMAN uses its own powerful job control and scripting language, as well as a published API that interfaces RMAN with many popular backup software solutions.

RMAN can store backups on the disk for quick recovery or place them on the tape for long-term storage. For RMAN to store backups on the tape, you must either use Oracle Secure Backup or configure an interface to the tape device known as a Media Management Library (MML).

Enterprise Manager supplies a graphical interface to the most commonly used RMAN functionality. Advanced backup and recovery operations are accessible through RMAN's command-line client. For more information about advanced RMAN capabilities, see the course titled *Oracle Database 11g: Administration Workshop II* or consult the *Oracle Backup and Recovery User's Guide*.

Configuring Backup Settings

Copyright © 2009, Oracle. All rights reserved.

Configuring Backup Settings

Select Enterprise Manager > Availability > Backup Settings. Here you can manage the persistent backup settings that are used for creating backups. There are separate settings for the disk and the tape. Tape settings depend on the media management library capabilities. Disk settings include:

- **Parallelism:** How many separate streams of backup information do you want to create? The best setting for parallelism depends on your hardware. As hardware resources increase, the appropriate degree of parallelism also increases. Generally, you want to set your parallelism to the number of disks that your disk backup location is striped over. For tape backup, you want to set your parallelism to the same number of tape drives that you have.
- **Disk backup location:** Where should backups be stored? The default is the fast recovery area. If you change this, click Test Disk Backup to verify that RMAN can write to the new location.
- **Disk backup type:** Select Backup Set, Compressed Backup Set, or Image Copy.

Click the Backup Set tab to set the maximum file size of backup pieces, specify the compression algorithm to be used for compressed backup sets, and specify redundancy for tape backups. Host credentials are required for Enterprise Manager to save changes to the backup settings.

Configuring Backup Settings

Best practice

Backup Policy

Automatically backup the control file and server parameter file (SPFILE) with every backup and database structural change

Autobackup Disk Location: An existing directory or diskgroup name where the control file and server parameter file will be backed up. If you do not specify a location, the files will be backed up to the fast recovery area location.

Optimize the whole database backup by skipping unchanged files such as read-only and offline datafiles that have been backed up

Enable block change tracking for faster incremental backups

Block Change Tracking File: Specify a location and file, otherwise an Oracle managed file will be created in the database area.

Tablespaces Excluded From Whole Database Backup

Populate this table with the tablespaces you want to exclude from a whole database backup. Use the Add button to add tablespace.

Select Tablespace Name	Tablespace Number	Status	Contents
No Items Selected			

TIP These tablespaces can be backed up separately using tablespace backup.

Retention Policy

Retain All Backups
You must manually delete any backups

Retain backups that are necessary for a recovery to any time within the specified number of days (point-in-time recovery)

Retain at least the specified number of full backups for each datafile

Days: 31
Recovery Window

Backups: 1
Redundancy

Archived Redo Log Deletion Policy

Specify the deletion policy for archived redo log files. The archived redo log files will be eligible for deletion if the fast recovery area becomes full.

None
If a fast recovery area is set, archived redo log files that have been backed up to a tertiary device and are obsolete based on the retention policy will be deleted.

Delete archived redo log files after they have been backed up the specified number of times
Backups: 1

Host Credentials

To save the backup settings, supply operating system login credentials to access the target database.

* Username:
* Password:
 Save as Preferred Credential

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Configuring Backup Settings (continued)

Click the Policy tab to:

- Automatically back up the control file and server parameter file (SPFILE) with each backup. You can also specify a location for these backups if you do not want them to go to the fast recovery area.
- Optimize backups by not backing up files that exactly match a file that is already part of the retained backups. This setting enables you to skip read-only and offline data files.
- Enable block change tracking and specify a location for the tracking file. If you intend to create incremental backups, this setting can decrease the time required to choose which blocks to include in the incremental backup.
- Exclude tablespaces from a whole database backup. Some administrators choose not to back up tablespaces containing data or objects that can be easily re-created (such as indexes or data that is batch-loaded frequently).
- Specify a retention policy: How long should RMAN keep your backups? If you are using the fast recovery area to store backups, RMAN automatically deletes old backups to make room for new ones (if the retention policy allows it). By default, only the last backup is retained. The retention policy can be specified as a number of backups or a number of days.

Scheduling Backups: Strategy

Schedule Backup

Oracle provides an automated backup strategy based on your disk and/or tape configuration. Alternatively, you can implement your own customized backup strategy.

Oracle-Suggested Backup

Schedule a backup using Oracle's automated backup strategy. [Schedule Oracle-Suggested Backup](#)

This option will back up the entire database. The database will be backed up on daily and weekly intervals.

Customized Backup

Select the object(s) you want to back up. [Schedule Customized Backup](#)

Whole Database
 Tablespaces
 Datafiles
 Archived Logs
 All Recovery Files on Disk
Includes all archived logs and disk backups that are not already backed up to tape.

Host Credentials

To perform a backup, supply operating system login credentials to access the target database.

* Username
* Password
 Save as Preferred Credential

Backup Strategies

Oracle-suggested:

- Provides an out-of-the-box backup strategy based on the backup destination
- Sets up recovery window for backup management
- Schedules recurring and immediate backups
- Automates backup management

Customized:

- Specify the objects to be backed up
- Choose disk or tape backup destination
- Override the default backup settings
- Schedule the backup

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Scheduling Backups: Strategy

Select Enterprise Manager > Availability > Schedule Backup. Select either the Oracle-Suggested Backup strategy or your own customized strategy. The Oracle-Suggested Backup strategy makes a one-time whole-database backup, which is performed online. This is a baseline incremental level 0 backup. The automated backup strategy then schedules incremental level 1 backups for each successive day. For day 3 and onward, RMAN applies the level 1 backup from the beginning of day n-1 to the level 0 backup before beginning the incremental backup for that day.

By clicking Schedule Customized Backup, you gain access to a wider range of configuration options. Select the objects that you want to back up—the whole database (the default) or individual tablespaces, data files, archived logs, or any Oracle backups currently residing on the disk (to move them to the tape).

Both strategies enable you to set up encrypted backups.

Scheduling Backups: Options

Options Settings Schedule Review

Schedule Customized Backup: Options

Database: orcl.oracle.com Backup Strategy: Customized Backup Object Type: Whole Database [Cancel](#) [Step 1 of 4](#) [Next](#)

Backup Type

Full Backup
 Use as the base of an incremental backup strategy

Incremental Backup
A level 1 cumulative incremental backup includes all blocks changed since the most recent level 0 backup.
 Refresh the latest datafile copy on disk to the current time using the incremental backup

Backup Mode

Online Backup
Can be performed when the database is open.

Offline Backup
If the database is open at the time of backup, it will be shut down and mounted before the backup, then re-opened after the backup.

Advanced

Also back up all archived logs on disk
 Delete all archived logs from disk after they are successfully backed up

Delete obsolete backups
Delete backups that are no longer required to satisfy the retention policy.

Use proxy copy supported by media management software to perform a backup
If proxy copy of the selected files is not supported, a conventional backup will be performed.

Maximum Files per Backup Set:

Section Size: KB [▼](#)
Backs up large files in parallel, using sections of the specified size. (This parameter overrides Maximum Backup Piece Size in Backup Settings.)

[►Encryption](#)

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Scheduling Backups: Options

Select full or incremental backup type. If you are performing a full database backup, you can select “Use as the base of an incremental backup strategy” to make the full database backup an incremental level 0 backup. If you are using image copies, you can select “Refresh the latest datafile copy on disk to the current time using the incremental backup” to update the existing backup rather than create a new image copy.

Select Online Backup if you want to perform this task while users are continuing to use the database. If users do not need access, select “Offline Backup,” which is performed with a mounted instance.

Select “Delete obsolete backups” to remove any backups that fall outside the retention policy that you configured earlier. RMAN automatically removes obsolete backups if you are backing up to the fast recovery area. Details about the advanced options and encryption are discussed in the course titled *Oracle Database 11g: Administration Workshop II* and in the backup and recovery documentation.

Scheduling Backups: Settings

Schedule Customized Backup: Settings

Database: **orcl.oracle.com** Backup Strategy: **Customized Backup** Object Type: **Whole Database**

Select the destination media for this backup. You can also override the default backup settings.

Disk Disk Backup Location: **+FRA**

Tape Media Management Vendor (MMV) Library Parameters: **Not specified**

[View Default Settings](#) [Override Default Settings](#)

Changed settings will only apply to the current backup.

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Scheduling Backups: Settings

Select whether the backup is to go to the disk or to the tape.

To create a one-time backup (in addition to your regularly scheduled backups), click **Override Current Settings** and specify your backup settings.

Scheduling Backups: Schedule

Options Settings **Schedule** Review

Schedule Customized Backup: Schedule

Database **orcl.oracle.com** Backup Strategy **Customized Backup** Object Type **Whole Database**

Job

* Job Name **BACKUP_ORCL.ORACLE.COM_000**
Job Description **Whole Database Backup**

Schedule

Type One Time (Immediately) One Time (Later) Repeating

Frequency Type **By Minutes**

Repeat Every Minutes

Time Zone **(UTC-08:00) US Pacific Time (PST)**

Start Date **Jun 18, 2009**

Start Time **1:00** AM PM

Repeat Until Indefinite
 Specified Date

Date **(example: Jun 18, 2009)**

Time AM PM

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Scheduling Backups: Schedule

Choose how you want the backup to be scheduled—either as a one-time job or as an automated, recurring process.

To configure a database for maximum recoverability, Oracle suggests regularly scheduled backups. Automating backups can simplify the administrator's workload.

When you select Repeating, the page displays additional scheduling details.

Scheduling Backups: Review

Schedule Customized Backup: Review

Database **orcl.oracle.com** Backup Strategy **Customized Backup** Object Type **Whole Database**

Cancel Edit RMAN Script Back Step 4 of 4 Submit Job

Settings

Destination	Disk
Backup Type	Full Backup
Backup Mode	Online Backup
Flash Recovery Area	+FRA

RMAN Script

The RMAN script below is generated based on previous input.

```
backup device type disk tag '%TAG' database;
backup device type disk tag '%TAG' archivelog all not backed up;
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Scheduling Backups: Review

RMAN uses its own command syntax and scripting language.

Using this page, you can customize the RMAN scripts (if needed) or copy them for recording purposes.

Backing Up the Control File to a Trace File

Control files have an additional backup option.

The screenshot shows the Oracle Enterprise Manager Control Files interface. On the left, there is a 'Control File Mirror Images' section with a table:

Valid	File Name	File Directory
VALID	current.260.689752023	+DATA/orcl/controlfile/
VALID	current.256.689752023	+FRA/orcl/controlfile/

On the right, there is a 'Control File Information' section with the following details:

- Database ID: 1217532758
- Control File Type: CURRENT
- Control File Creation Date: June 17, 2009 5:47:05 AM
- Control File Sequence Number: 1557
- Last Change Number: 1025918
- Date Last Modified: June 18, 2009 12:13:30 PM
- Control File AutoBackup: Enabled [Click here to disable](#)

Control file trace backups
may be used to recover
from loss of all control files.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Backing Up the Control File to a Trace File

Select Enterprise Manager > Server > Control Files to manage your database's control files. Control files have an additional backup option; they may be backed up to a trace file. A control file trace backup contains the SQL statement required to re-create the control files in the event that all control files are lost.

Although it is very unlikely that a properly configured database (with multiple copies of the control file placed on separate disks and separate controllers) would lose all control files at the same time, it is possible. Therefore, the administrator should back up the control file to a trace file after each change to the physical structure of the database (adding tablespaces or data files, or adding additional redo log groups).

Trace copies of the control file can be created by using Enterprise Manager (as shown in the slide) or with the following SQL command:

```
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE;
```

The trace backup is created in the location specified by the `DIAGNOSTIC_DEST` initialization parameter (for example, `/u01/app/oracle/diag/rdbms/orcl/orcl/trace` with a file name such as `orcl_vktm_8400.trc`).

You can view information from within the control file on the Advanced tab of the Control Files page.

Managing Backups

Select	Key	Tag	Completion Time	Contents	Device Type	Status	Keep	Pieces
<input type="checkbox"/>	4	TAG20090618T121325	Jun 18, 2009 12:13:27 PM	CONTROLFILE, SPFILE	DISK	AVAILABLE	NO	1
<input type="checkbox"/>	3	BACKUP_ORCL_ORACLE_061809120854	Jun 18, 2009 12:13:21 PM	ARCHIVED LOG	DISK	AVAILABLE	NO	1
<input type="checkbox"/>	2	TAG20090618T121228	Jun 18, 2009 12:12:56 PM	CONTROLFILE, SPFILE	DISK	AVAILABLE	NO	1
<input type="checkbox"/>	1	BACKUP_ORCL_ORACLE_061809120854	Jun 18, 2009 12:12:20 PM	DATAFILE	DISK	AVAILABLE	NO	1

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Managing Backups

Select Enterprise Manager > Availability > Manage Current Backup to manage your existing backups. On this page, you can see when a backup was completed, where it was created (disk or tape), and whether it is still available.

At the top of the Manage Current Backups page, four buttons enable you to work with existing backups:

- **Catalog Additional Files:** Although RMAN (working through Enterprise Manager) is the recommended way to create backups, you might have image copies or backup sets that were created by some other means or in some other environment with the result that RMAN is not aware of them. This task identifies those files and adds them to the catalog.
- **Crosscheck All:** RMAN can automatically delete obsolete backups, but you can also delete them by using operating system commands. If you delete a backup without using RMAN, the catalog does not know whether the backup is missing until you perform a cross-check between the catalog and what is really there.
- **Delete All Obsolete:** This deletes backups older than the retention policy.
- **Delete All Expired:** This deletes the catalog listing for any backups that are not found when the cross-check is performed.

Viewing Backup Reports

[View Backup Report](#)

The following backup jobs are known to the database. The data is retrieved from the database control file.

Search

Status: All Start Time: Within 1 month Type: All [Go](#)

Results

Total 1 (Completed ✓ 1)

Backup Name	Status	Start Time	Time Taken	Type	Output Devices	Input Size	Output Size	Output Rate (Per Sec)
BACKUP_ORCL.ORACLE_061809120854	COMPLETED	Jun 18, 2009 12:09:16	00:04:14	DB	DISK	1.64G	1.34G	5.42M

Related Links

[Manage Current](#)

Inputs

Datafiles

Datafile Number	Output Type	Output Key	File Size	Tablespace	Checkpoint Time	Incremental Level	Compression Ratio	Corrupted Blocks	File Creation Time	File Checkpoint SCN	Resetlogs SCN
1	BACKUPSET	1	750.01M	SYSTEM	Jun 18, 2009 12:09:18 PM GMT+07:00		1.158	0	Apr 24, 2009 10:31:11 AM GMT+07:00	1025302	740137
2	BACKUPSET	1	651.26M	SYSAUX	Jun 18, 2009 12:09:18 PM GMT+07:00		1.43	0	Apr 24, 2009 10:31:17 AM GMT+07:00	1025302	740137
3	BACKUPSET	1	100.01M	UNDOTBS1	Jun 18, 2009 12:09:18 PM GMT+07:00		11.595	0	Apr 24, 2009 11:29:42 AM GMT+07:00	1025302	740137
4	BACKUPSET	1	148.20M	USERS	Jun 18, 2009 12:09:18 PM GMT+07:00		1.102	0	Apr 24, 2009 10:31:30 AM GMT+07:00	1025302	740137
5	BACKUPSET	1	100.01M	EXAMPLE	Jun 18, 2009 12:09:18 PM GMT+07:00		1.442	0	Jun 17, 2009 5:49:29 AM GMT+07:00	1025302	740137

Control Files

Output Type	Output Key/Checkpoint Time	File Size	File Checkpoint SCN	Resetlogs SCN
BACKUPSET	2 Jun 18, 2009 12:12:28 PM GMT+07:00	9.30M	1025706	740137
BACKUPSET	4 Jun 18, 2009 12:13:25 PM GMT+07:00	9.30M	1025894	740137

SPFile

Backup Set Modification Time	File Size
4 Jun 18, 2009 11:21:10 AM GMT+07:00	0.00K
2 Jun 18, 2009 11:21:10 AM GMT+07:00	0.00K

Archived Logs

Output Type	Output Key	Thread Number	Sequence Number	File Size	Low Time	High Time	Compression Ratio	Resetlogs SCN
BACKUPSET	3	1	14	42.14M	Jun 18, 2009 6:41:27 AM GMT+07:00	Jun 18, 2009 12:13:04 PM GMT+07:00	1	740137

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Viewing Backup Reports

Information about backup jobs can also be viewed by selecting Enterprise Manager > Availability > Backup Reports. The content is based on the control file. The backup report contains summary information as well as detailed information about the input and output of a specific job, including timing, SCN, sizing, compression, corruption (if any), and so on.

Monitoring the Fast Recovery Area

Flash Recovery

This database is using a flash recovery area. The chart shows space used by each file type that is not reclaimable by Oracle. Performing backups to tertiary storage is one way to make space reclaimable. Usable Flash Recovery Area includes free and reclaimable space.

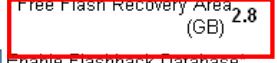
Flash Recovery Area Location 

Flash Recovery Area Size MB 

Flash Recovery Area Size must be set when the location is set.

Non-reclaimable Flash Recovery Area (GB) **1.5**

Reclaimable Flash Recovery Area (MB) **53**

Free Flash Recovery Area (GB) **2.8** 

Enable Flashback Database
Flashback database can be used for fast database point-in-time recovery, as it returns the database to a prior point-in-time without restoring files. Flashback is the preferred point-in-time recovery method in the recovery wizard when appropriate. The flash recovery area must be set to enable flashback database.

Flashback Retention Time Hours 

Current size of the flashback logs(GB) **n/a**

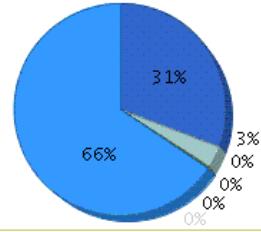
Lowest SCN in the flashback data **n/a**

Flashback Time **n/a**

Apply initialization parameter changes to SPFILE only. If not checked, parameter changes will be made to both the SPFILE and the running instance.

* Changes to this setting or parameter require a database restart.

Flash Recovery Area Usage



File Type	Size (GB)	Percentage
Backup Piece	1.34	30.8%
Online Log	0.15	3.4%
Control File	0.01	0.2%
Archived Redo Log	0	0%
Image Copy	0	0%
Flashback Log	0	0%
Usable	2.85	65.6%

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Monitoring the Fast Recovery Area

If you have configured your archived logs to be written to this location, it is important to monitor this space to ensure that it does not reach its capacity. If the instance is unable to create an archived log because of lack of space, it pauses until the administrator corrects the situation.

Select Enterprise Manager > Availability > Recovery Settings. On this page, you can:

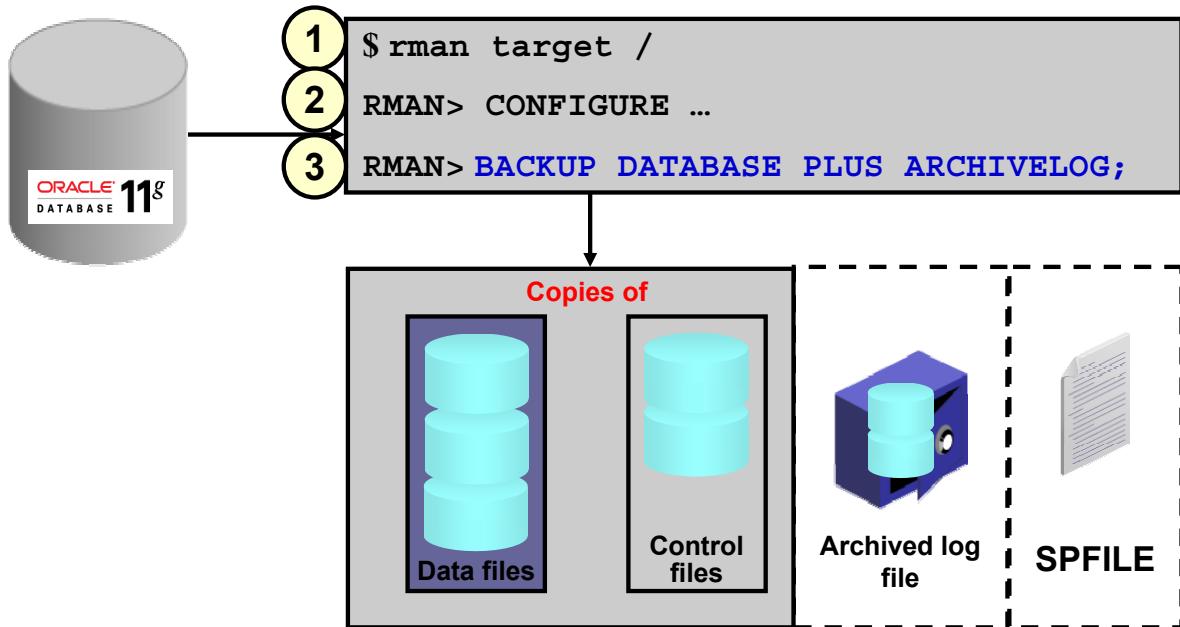
- Verify how much of the fast recovery area has been consumed
- Specify the location of the fast recovery area
- Specify the size of the fast recovery area
- Configure Flashback Database
- Specify the retention time

The retention time determines when files are obsolete (that is, when they are no longer needed to meet your data recovery objectives). The Oracle database automatically manages this storage, deleting files that are no longer needed. When you back up the recovery area, RMAN can fail over to other archived redo log destinations if the archived redo log in the fast recovery area is inaccessible or corrupted.

Periodically copying backups to tape frees space in the fast recovery area for other files, but retrieving files from tape causes longer database restoration and recovery times.

Note: The Enterprise Manager interface has not yet been updated to reflect the change in name from flash recovery area to fast recovery area.

Using the RMAN Command Line



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using the RMAN Command Line

1. In a terminal session, start RMAN and connect to the target database.
2. Execute configuration commands:
 - CONFIGURE DEFAULT DEVICE TYPE TO disk;
 - CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO COPY;
 - CONFIGURE CONTROLFILE AUTOBACKUP ON;
3. A whole database backup is a copy of all data files and the control file. You can optionally include the server parameter file (SPFILE) and archived redo log files. Using RMAN to make an image copy of all the database files simply requires mounting or opening the database, starting RMAN, and entering the BACKUP command shown in the slide. Optionally, you can supply the DELETE INPUT option when backing up archive log files. That causes RMAN to remove the archive log files after backing them up. This is useful especially if you are not using a fast recovery area, which would perform space management for you, deleting files when space pressure grows. In that case, the command in the slide would look like the following:

RMAN> BACKUP DATABASE PLUS ARCHIVELOG DELETE INPUT;

You can also create a backup (either a backup set or image copies) of previous image copies of all data files and control files in the database by using the following command:

RMAN> BACKUP COPY OF DATABASE;

Quiz

Using the change-tracking feature, an image copy backup performed by RMAN can skip blocks that have not changed since the last backup.

1. True
2. False

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answer: 2

Summary

In this lesson, you should have learned how to:

- Create consistent database backups
- Back up your database without shutting it down
- Create incremental backups
- Automate database backups
- Manage backups and view backup reports
- Monitor the fast recovery area



Copyright © 2009, Oracle. All rights reserved.

Practice 15 Overview: Creating Database Backups

This practice covers the following topics:

- Backing up your database while the database is open for user activity
- Scheduling automatic nightly incremental backups for your database



Copyright © 2009, Oracle. All rights reserved.

16

Performing Database Recovery

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Determine the need for performing recovery
- Access different interfaces (such as Enterprise Manager and command line)
- Describe and use available options, such as Recovery Manager (RMAN) and the Data Recovery Advisor
- Perform recovery:
 - Control file
 - Redo log file
 - Data file

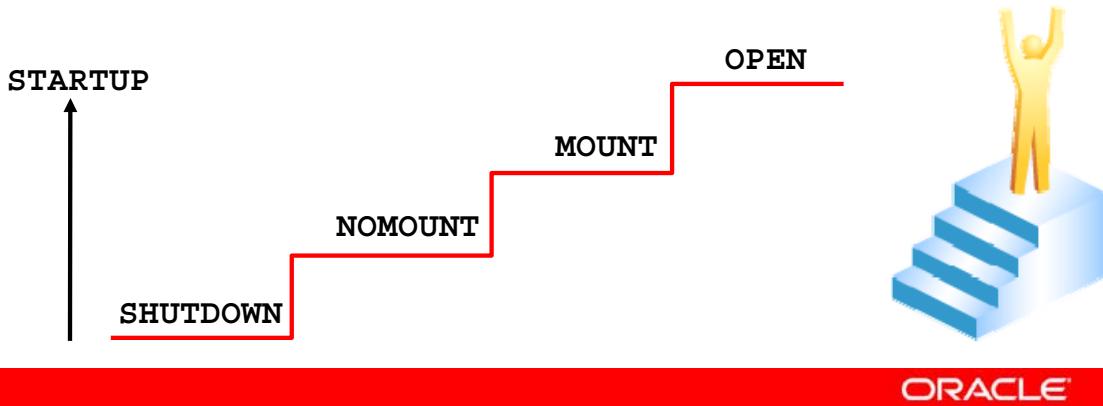
ORACLE

Copyright © 2009, Oracle. All rights reserved.

Opening a Database

To open a database:

- All control files must be present and synchronized
- All online data files must be present and synchronized
- At least one member of each redo log group must be present



Opening a Database

As a database moves from the shutdown stage to being fully open, it performs internal consistency checks with the following stages:

- **NOMOUNT:** For an instance to reach the NOMOUNT (also known as STARTED) status, the instance must read the initialization parameter file. No database files are checked while the instance enters the NOMOUNT state.
- **MOUNT:** As the instance moves to the MOUNT status, it checks whether all control files listed in the initialization parameter file are present and synchronized. If even one control file is missing or corrupt, the instance returns an error (noting the missing control file) to the administrator and remains in the NOMOUNT state.
- **OPEN:** When the instance moves from the MOUNT state to the OPEN state, it does the following:
 - Checks whether all redo log groups known to the control file have at least one member present. Any missing members are noted in the alert log.

Opening a Database (continued)

- Verifies that all data files known to the control file are present unless they have been taken offline. Offline files are not checked until the administrator tries to bring them online. The administrator may take a data file offline and open the instance if the data file does not belong to the SYSTEM or UNDO tablespaces. If any files are missing, an error noting the first missing file is returned to the administrator and the instance remains in the MOUNT state. When the instance finds files that are missing, only the first file causing a problem appears in the error message. To find all files that need recovery, the administrator can check the v\$recover_file dynamic performance view to get a complete list of the files that need attention:

```
SQL> startup
ORACLE instance started.
Total System Global Area  171966464 bytes
  Fixed Size                  775608 bytes
  Variable Size              145762888 bytes
  Database Buffers            25165824 bytes
  Redo Buffers                262144 bytes
Database mounted.
ORA-01157: cannot identify/lock data file 4 - see DBWR trace
file
ORA-01110: data file 4: '/oracle/oradata/orcl/users01.dbf'
SQL> SELECT name, error
  2  FROM v$datafile
  3  JOIN v$recover_file
  4  USING (file#);
NAME                           ERROR
-----
/oracle/oradata/orcl/users01.dbf  FILE NOT FOUND
/oracle/oradata/orcl/example01.dbf FILE NOT FOUND
```

- Verifies that all data files that are not offline or read-only are synchronized with the control file. If necessary, instance recovery is automatically performed. However, if a file is out of synchronization to the extent that it cannot be recovered by using the online redo log groups, then the administrator must perform media recovery. If any files require media recovery, an error message noting the first file requiring recovery is returned to the administrator and the instance remains in the MOUNT state:

```
ORA-01113: file 4 needs media recovery
ORA-01110: data file 4: '/oracle/oradata/orcl/users01.dbf'
```

Again, v\$recover_file gives a complete list of files that need attention. Files that are present and require media recovery are listed, but no error message is displayed.

Keeping a Database Open

After the database is open, it fails in the case of the loss of:

- Any control file
- A data file belonging to the system or undo tablespaces
- An entire redo log group
(As long as at least one member of the group is available, the instance remains open.)

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Keeping a Database Open

After a database is open, instance failure can be caused by media failure: for example, by the loss of a control file, the loss of an entire redo log group, or the loss of a data file belonging to the SYSTEM or UNDO tablespaces. Even if an inactive redo log group is lost, the database would eventually fail due to log switches.

In many cases, the failed instance does not completely shut down but is unable to continue to perform work. Recovering from these types of media failure must be done with the database down. As a result, the administrator must use the SHUTDOWN ABORT command before beginning recovery efforts.

The loss of data files belonging to other tablespaces does not cause instance failure, and the database can be recovered while open, with work continuing in other tablespaces.

These errors can be detected by inspecting the alert log file or by using the Data Recovery Advisor.

Data Recovery Advisor

- Fast detection, analysis, and repair of failures
- Down-time and run-time failures
- Minimizing disruptions for users
- User interfaces:
 - Enterprise Manager GUI (several paths)
 - RMAN command line
- Supported database configurations:
 - Single instance
 - Not RAC
 - Supporting failover to standby, but not analysis and repair of standby databases



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Recovery Advisor

The Data Recovery Advisor automatically gathers data failure information when an error is encountered. In addition, it can proactively check for failures. In this mode, it can potentially detect and analyze data failures before a database process discovers the corruption and signals an error. (Note that repairs are always under human control.)

Data failures can be very serious. For example, if your current log files are missing, you cannot open your database. Some data failures (like block corruptions in data files) are not catastrophic because they do not take the database down or prevent you from opening the Oracle database. The Data Recovery Advisor handles both cases: the one when you cannot start up the database (because required database files are missing, inconsistent, or corrupted) and the one when file corruptions are discovered during run time.

The preferred way to address serious data failures is as follows:

1. Fail over to a standby database if you are in a Data Guard configuration. This allows users to come back online as soon as possible.
2. Repair the primary cause of the data failure (fortunately, this does not affect your users).

Functionality of the Data Recovery Advisor (continued)

User Interfaces

The Data Recovery Advisor is available from Enterprise Manager (EM) Database Control and Grid Control. When failures exist, there are several ways to access the Data Recovery Advisor. The following examples all begin on the Database Instance home page:

- Availability tabbed page > Perform Recovery > Advise and Recover
- Active Incidents link > on the Support Workbench “Problems” page: Checker Findings tabbed page > Launch Recovery Advisor
- Database Instance Health > click specific link (for example, ORA 1578) in the Incidents section > Support Workbench, Problems Detail page > Data Recovery Advisor
- Database Instance Health > Related Links section: Support Workbench > Checker Findings tabbed page: Launch Recovery Advisor
- Related Link: Advisor Central > Advisors tabbed page: Data Recovery Advisor
- Related Link: Advisor Central > Checkers tabbed page: Details > Run Detail tabbed page: Launch Recovery Advisor

You can also use the Data Recovery Advisor by using the RMAN command line:

```
rman target /  
rman> list failure all;
```

Supported Database Configurations

In the current release, the Data Recovery Advisor supports single-instance databases. Oracle Real Application Clusters databases are not supported.

The Data Recovery Advisor cannot use blocks or files transferred from a standby database to repair failures on a primary database. Furthermore, you cannot use the Data Recovery Advisor to diagnose and repair failures on a standby database. However, the Data Recovery Advisor does support failover to a standby database as a repair option (as mentioned above).

Loss of a Control File

If a control file is lost or corrupted, the instance normally aborts.

- If control files are stored in ASM disk groups, recovery options are as follows:
 - Perform guided recovery using Enterprise Manager.
 - Put database in NOMOUNT mode and use an RMAN command to restore control file from existing control file.

```
RMAN> restore controlfile from
      '+DATA/orcl/controlfile/current.260.695209463' ;
```

- If control files are stored as regular file system files then:
 - Shut down the database
 - Copy existing control file to replace lost control file

After control file is successfully restored, open the database.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Loss of a Control File

The options for recovery from the loss of a control file depend on the storage configuration of the control files and on whether at least one control file remains or have all been lost.

If using ASM storage, and at least one control file copy remains, you can perform guided recovery using Enterprise Manager or perform manual recovery using RMAN as follows:

1. Put the database in NOMOUNT mode.
2. Connect to RMAN and issue the `restore controlfile` command to restore the control file from an existing control file, for example:
`restore controlfile from
 '+DATA/orcl/controlfile/current.260.695209463' ;`
3. After the control file is successfully restored, open the database.

If your control files are stored as regular file system files and at least one control file copy remains, then, while the database is down, you can just copy one of the remaining control files to the missing file's location. If the media failure is due to the loss of a disk drive or controller, copy one of the remaining control files to some other location and update the instance's parameter file to point to the new location. Alternatively, you can delete the reference to the missing control file from the initialization parameter file. Remember that Oracle recommends having at least two control files at all times.

Note: Recovering from the loss of all control files is covered in the course titled *Oracle Database 11g: Administration Workshop II*.

Loss of a Redo Log File

If a member of a redo log file group is lost and if the group still has at least one member, note the following results:

- Normal operation of the instance is not affected.
- You receive a message in the alert log notifying you that a member cannot be found.
- You can restore the missing log file by dropping the lost redo log member and adding a new member.
- If the group with the missing log file has been archived you can clear the log group to re-create the missing file.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Loss of a Redo Log File

Recovering from the loss of a single redo log group member should not affect the running instance.

To perform this recovery:

1. Determine whether there is a missing log file by examining the alert log.
2. Restore the missing file by first dropping the lost redo log member:

```
SQL> ALTER DATABASE DROP LOGFILE MEMBER  
'+DATA/orcl/onlinelog/group_1.261.691672257';
```

Then add a new member to replace the lost red log member:

```
SQL> ALTER DATABASE ADD LOGFILE MEMBER '+DATA' TO GROUP 2;
```

Enterprise Manager can also be used to drop and re-create the log file member.

Note: If using OMF for your redo log files and you use the above syntax to add a new redo log member to an existing group, that new redo log member file will not be an OMF file. If you want to ensure that the new redo log member is an OMF file, then the easiest recovery option would be to create a new redo log group and then drop the redo log group that had the missing redo log member.

3. If the media failure is due to the loss of a disk drive or controller, rename the missing file.
4. If the group has already been archived, or if you are in NOARCHIVELOG mode, you may choose to solve the problem by clearing the log group to re-create the missing file or files. Select the appropriate group and then select the Clear Logfile action. You can also clear the affected group manually with the following command:

```
SQL> ALTER DATABASE CLEAR LOGFILE GROUP #;
```

Loss of a Redo Log File (continued)

Note: Database Control does not allow you to clear a log group that has not been archived. Doing so breaks the chain of redo information. If you must clear an unarchived log group, you should *immediately* take a full backup of the whole database. Failure to do so may result in a loss of data if another failure occurs. To clear an unarchived log group, use the following command:

```
SQL> ALTER DATABASE CLEAR UNARCHIVED LOGFILE GROUP #;
```

Loss of a Data File in NOARCHIVELOG Mode

If the database is in NOARCHIVELOG mode and if any data file is lost, perform the following tasks:

1. Shut down the instance if it is not already down.
2. Restore the entire database—including all data and control files—from the backup.
3. Open the database.
4. Have users reenter all changes that were made since the last backup.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Loss of a Data File in NOARCHIVELOG Mode

The loss of *any* data file from a database in NOARCHIVELOG mode requires complete restoration of the database, including control files and all data files.

With the database in NOARCHIVELOG mode, recovery is possible only up to the time of the last backup. So users must reenter all changes made since that backup.

To perform this type of recovery:

1. Shut down the instance if it is not already down.
2. Click Perform Recovery on the Maintenance properties page.
3. Select Whole Database as the type of recovery.

If you have a database in NOARCHIVELOG mode that has an incremental backup strategy, RMAN first restores the most recent level 0 and then RMAN recovery applies the incremental backups.

Loss of a Noncritical Data File in ARCHIVELOG Mode

If a data file is lost or corrupted, and if that file does not belong to the SYSTEM or UNDO tablespace, you restore and recover the missing data file.

Object Level Recovery

Object Type

Operation Type Recover to current time
Datafile will be restored as required.
 Restore datafiles
Specify Time, SCN or log sequence. The backup taken at or prior to that time will be used. No recovery will be performed in this operation.
 Recover from previously restored datafiles
 Block Recovery



Users

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Loss of a Noncritical Data File in ARCHIVELOG Mode

With the database in ARCHIVELOG mode, the loss of any data file not belonging to the SYSTEM or UNDO tablespaces affects only the objects that are in the missing file. The rest of the database remains available for users to continue work.

To restore and recover the missing data file:

1. Click Perform Recovery on the Maintenance properties page.
2. Select Datafiles as the recovery type, and then select “Restore to current time.”
3. Add all data files that need recovery.
4. Determine whether you want to restore the files to the default location or (if a disk or controller is missing) to a new location.
5. Submit the RMAN job to restore and recover the missing files.

Because the database is in ARCHIVELOG mode, recovery is possible up to the time of the last commit and users are not required to reenter any data.

Loss of a System-Critical Data File in ARCHIVELOG Mode

If a data file is lost or corrupted, and if that file belongs to the SYSTEM or UNDO tablespace, perform the following tasks:

1. The instance may or may not shut down automatically. If it does not, use SHUTDOWN ABORT to bring the instance down.
2. Mount the database.
3. Restore and recover the missing data file.
4. Open the database.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Loss of a System-Critical Data File in ARCHIVELOG Mode

Data files belonging to the SYSTEM tablespace or containing UNDO data are considered system critical. A loss of one of these files requires the database to be restored from the MOUNT state (unlike other data files that may be restored with the database open).

To perform this recovery:

1. If the instance is not already shut down, shut it down.
2. Mount the database.
3. Click Perform Recovery on the Maintenance properties page.
4. Select Datafiles as the recovery type, and then select “Restore to current time.”
5. Add all data files that need recovery.
6. Determine whether you want to restore the files to the default location or (if a disk or controller is missing) to a new location.
7. Submit the RMAN job to restore and recover the missing files.
8. Open the database. Users are not required to reenter data because the recovery is up to the time of the last commit.

Data Failure: Examples

- Inaccessible components: Missing data files at the OS level, incorrect access permissions, offline tablespace
- Physical corruptions: Block checksum failures, invalid block header field values
- Logical corruptions: Inconsistent dictionary; corrupt row piece, index entry, or transaction
- Inconsistencies: Control file older or newer than the data files and online redo logs
- I/O failures: Limit on the number of open files exceeded, inaccessible channels, network or I/O error



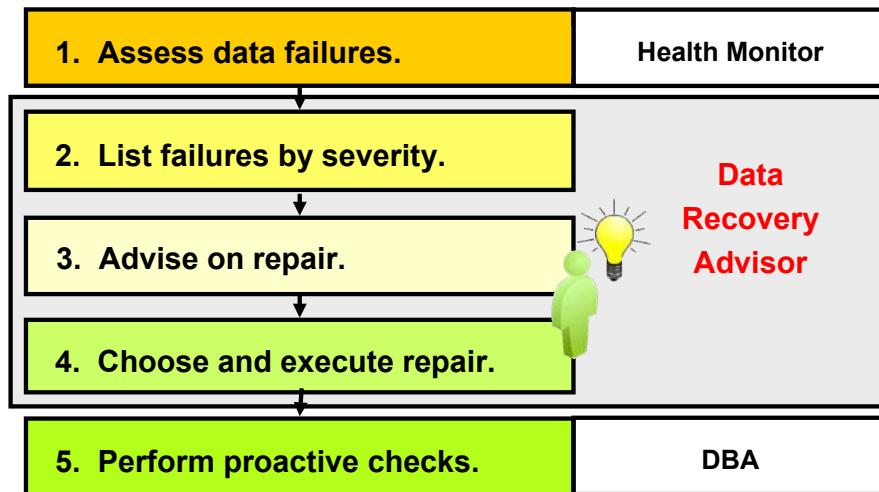
ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Data Failure: Examples

The Data Recovery Advisor can analyze failures and suggest repair options for a growing list of issues.

Data Recovery Advisor



ORACLE

Copyright © 2009, Oracle. All rights reserved.

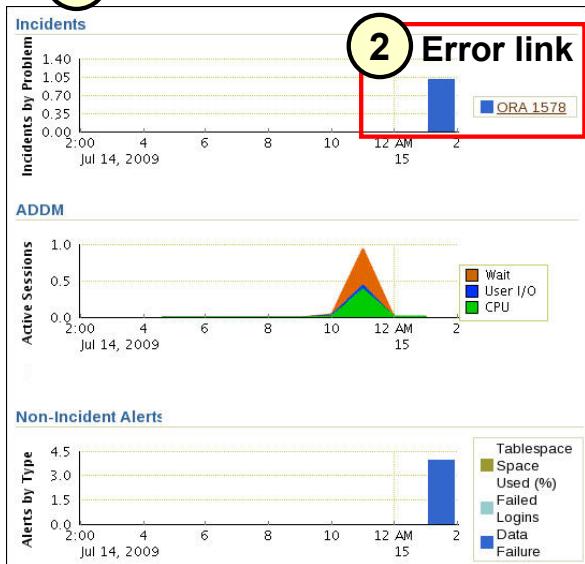
Data Recovery Advisor

The automatic diagnostic workflow in Oracle Database 11g performs workflow steps for you. With the Data Recovery Advisor you only need to initiate an advise and a repair.

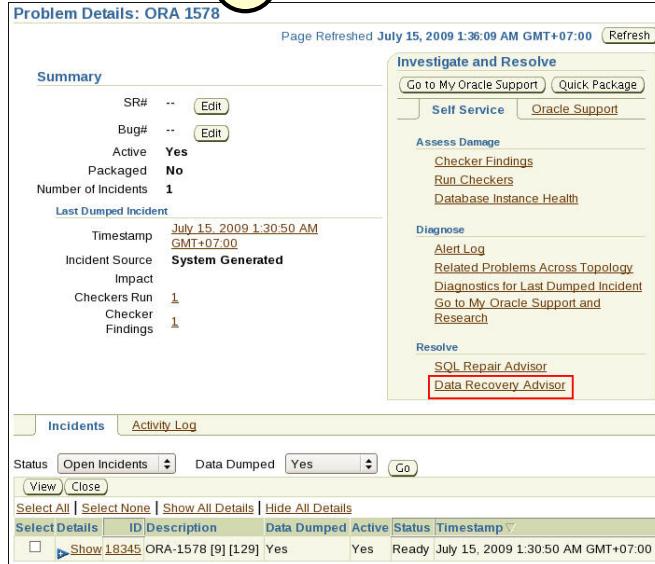
1. Health Monitor automatically executes checks and logs failures and their symptoms as “findings” into the automatic diagnostic repository (ADR).
2. The Data Recovery Advisor consolidates findings into failures. It lists the results of previously executed assessments with failure severity (critical or high).
3. When you ask for repair advice on a failure, the Data Recovery Advisor maps failures to automatic and manual repair options, checks basic feasibility, and presents you with the repair advice.
4. You can execute a repair manually, or you can request the Data Recovery Advisor to do it for you.
5. In addition to the automatic, primarily “reactive” checks of Health Monitor and the Data Recovery Advisor, Oracle recommends using the VALIDATE command as a “proactive” check.

Assessing Data Failures

1 Database instance health



3 Problem details



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Assessing Data Failures

The example in the slide shows one of several possible ways to see the interaction of Health Monitor and the Data Recovery Advisor.

Data Failures

The screenshot shows the Oracle Enterprise Manager 11g Database Control interface. The top navigation bar includes 'ORACLE Enterprise Manager 11g', 'Database Control', 'Help', and 'Database'. The main content area has a green header bar with the text 'Database Instance: orcl.oracle.com > Information'. Below this, a list of failures is shown: 1. Database Failures - 1 and 2. Current Status - MOUNTED. A 'Perform Recovery' section follows, with a sub-section 'Oracle Advised Recovery' containing a message from the Data Recovery Advisor and a 'Advise and Recover' button. Another sub-section, 'User Directed Recovery', shows a 'Recovery Scope' dropdown set to 'Whole Database' and a 'Recover' button. The 'Operation Type' section contains three radio buttons: 'Recover to the current time or a previous point-in-time' (selected), 'Restore all datafiles', and 'Recover from previously restored datafiles'. To the right of these sections is a 'Overview' box with a list of recovery options:

- Recover database failures as advised by Oracle
- Restore and/or recover the entire database or selected objects
- Restore files to a new location
- Recover tablespaces to a point-in-time based on a timestamp, system change number (SCN), or log sequence number
- Recover datafile data blocks that are marked as corrupted, or based on datafile block IDs or tablespace block addresses
- Flashback database or tables to a specific system change number (SCN) or timestamp

At the bottom of the interface is a red bar with the 'ORACLE' logo and the text 'Copyright © 2009, Oracle. All rights reserved.'

Data Failures

Data failures are detected by checks, which are diagnostic procedures that assess the health of the database or its components. Each check can diagnose one or more failures, which are then mapped to a repair.

Checks can be reactive or proactive. When an error occurs in the database, reactive checks are automatically executed. You can also initiate proactive checks (for example, by executing the `VALIDATE DATABASE` command).

In Enterprise Manager, select Availability > Perform Recovery or click the Perform Recovery button if you find your database in a “down” or “mounted” state. Click “Advise and Recover” to have Enterprise Manager analyze and produce recovery advice.

Listing Data Failures

Select dropdown values and optionally enter failure description and impact strings to filter the data that is displayed in your results set.

Failure Description	Impact	Priority	Status	Time Detected
	CRITICAL or HIGH	OPEN	All	Go

Select failures and ...

Select All | Select None | Expand All | Collapse All

Select Failure Description	Impact	Priority	Status	Time Detected
<input type="checkbox"/> Data Failures				
<input checked="" type="checkbox"/> System datafile 1: '+DATA/orcl /datafile/system.256.692202091' is missing	Database cannot be opened	CRITICAL	OPEN	2009-07-15 02:36:06.0

TIP All CRITICAL failures must be selected before "Advise". All CRITICAL failures must be unselected before "Set Priority High" or "Set Priority Low".

Use a Recovery Catalog

Recovery Catalog Database **Not specified**

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Listing Data Failures

This “View and Manage Failures” page is the home page for the Data Recovery Advisor. The example in the screenshot shows how the Data Recovery Advisor lists data failures and details. Activities that you can initiate include advising, setting priorities, and closing failures.

The underlying RMAN LIST FAILURE command can also display data failures and details. Failure assessments are not initiated here; they are executed and stored in the ADR.

Failures are listed in decreasing priority order: CRITICAL, HIGH, LOW. Failures with the same priority are listed in order of increasing time stamps.

Advising on Repair

The screenshot shows the Oracle Enterprise Manager 11g Database Control interface. On the left, the 'Manual Actions' page is displayed. It features a '1' icon with 'Re-assess Failures' and a '2a' icon with 'Continue with Advise'. A blue box highlights the 'Continue with Advise' button. Below these are 'Manual Action Details' and an 'RMAN Script' section. An arrow points from the 'RMAN Script' section to the 'Review' page on the right. The 'Review' page has a '2b' icon with 'Execute Recovery'. A red box highlights the 'Execute Recovery' button. The 'Review' page also shows a table of 'Failures That Will Be Resolved' with one entry: 'System datafile 1: '+DATA/orcl/datafile/system.256.692202091' is missing'.

(1) After manual repair
(2) Automatic repair

Advising on Repair

On the “View and Manage Failures” page, the Data Recovery Advisor generates a manual checklist after you click the Advise button. Two types of failures can appear.

- Failures that require human intervention: An example is a connectivity failure when a disk cable is not plugged in.
- Failures that are repaired faster if you can undo a previous erroneous action: For example, if you renamed a data file by error, it is faster to rename it back to its previous name than to initiate RMAN restoration from backup.

You can initiate the following actions:

- Click “Re-assess Failures” after you perform a manual repair. Resolved failures are implicitly closed; any remaining failures are displayed on the “View and Manage Failures” page.
- Click “Continue with Advise” to initiate an automated repair. When the Data Recovery Advisor generates an automated repair option, it generates a script that shows how RMAN plans to repair the failure. Click Continue if you want to execute the automated repair. If you do not want the Data Recovery Advisor to automatically repair the failure, you can use this script as a starting point for your manual repair.

Executing Repairs

Database Instance: orcl.oracle.com >

Recovery Succeeded
Recovery succeeded. See Recovery Results below.

Recovery Results

Recovery Manager: Release 11.2.0.1.0 - Production on Wed Jul 15 02:47:57 2009

Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.

RMAN>
connected to target database: ORCL (DBID=1219972082, not open)
using target database control file instead of recovery catalog

RMAN> echo set on

media recovery complete, elapsed time **00:00:40**
Finished recover at 15-JUL-09
repair failure complete

RMAN> exit;

Recovery Manager complete.

Open Database **OK**

Open Database **OK**

i The database has been opened successfully.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Executing Repairs

The Data Recovery Advisor displays these pages. In the example, a successful repair is completed in 40 seconds.

Data Recovery Advisor Views

Querying dynamic data dictionary views

- **V\$IR_FAILURE:** Listing of all failures, including closed ones (result of the LIST FAILURE command)
- **V\$IR_MANUAL_CHECKLIST:** Listing of manual advice (result of the ADVISE FAILURE command)
- **V\$IR_REPAIR:** Listing of repairs (result of the ADVISE FAILURE command)
- **V\$IR_FAILURE_SET:** Cross-reference of failure and advise identifiers



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Recovery Advisor Views

Usage Example

Suppose that you need to display all failures that were detected on June 21, 2007.

```
SELECT * FROM v$ir_failure
  WHERE trunc (time_detected) = '21-JUN-2007';
```

See the *Oracle Database Reference* for details about the dynamic data dictionary views used by the Data Recovery Advisor.

Quiz

An Instance will not fail if the following event occurs:

1. Loss of a control file if there is a remaining multiplexed control file
2. Loss of the SYSTEM tablespace
3. Loss of one redo log member if there is a remaining multiplexed redo log member from the same group of the lost member
4. Loss of the active undo tablespace

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answer: 3

Quiz

The information used by the Data Recovery Advisor is only available via the Enterprise Manager interface.

1. True
2. False



Copyright © 2009, Oracle. All rights reserved.

Answer: 2

Summary

In this lesson, you should have learned how to:

- Determine the need for performing recovery
- Access different interfaces (such as Enterprise Manager and command line)
- Describe and use available options, such as Recovery Manager (RMAN) and the Data Recovery Advisor
- Perform recovery:
 - Control file
 - Redo log file
 - Data file

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Practice 16 Overview: Performing Database Recovery

This practice covers recovering from the loss of a:

- Control file
- Noncritical data file
- System-critical data file



Copyright © 2009, Oracle. All rights reserved.

17

Moving Data

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Objectives

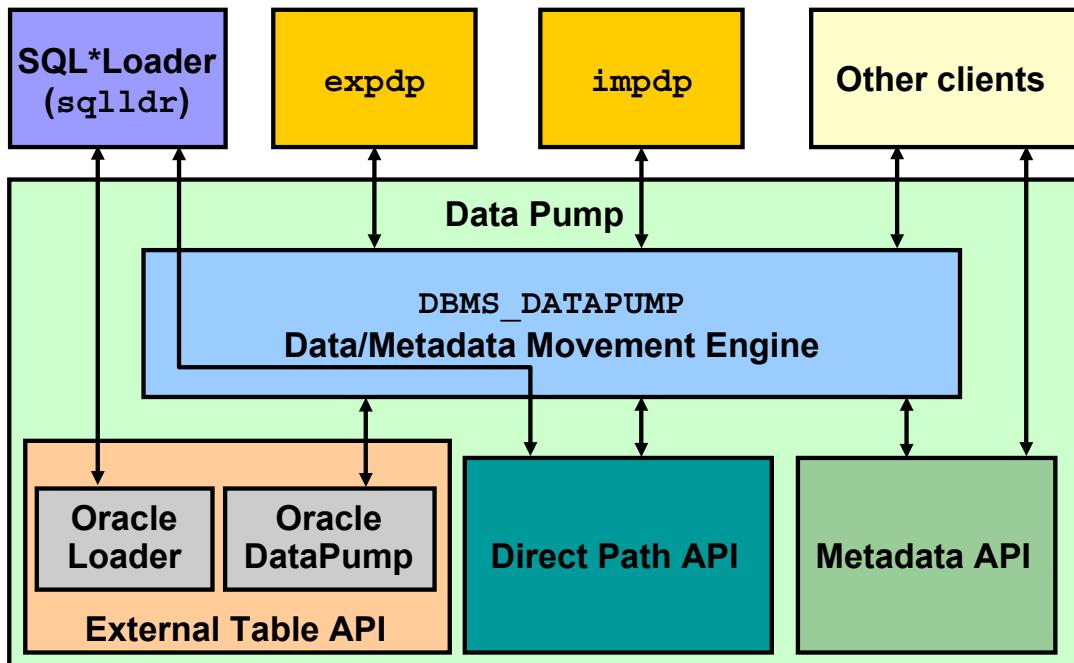
After completing this lesson, you should be able to:

- Describe ways to move data
- Create and use directory objects
- Use SQL*Loader to load data from a non-Oracle database (or user files)
- Use external tables to move data via platform-independent files
- Explain the general architecture of Oracle Data Pump
- Use Data Pump Export and Import to move data between Oracle databases

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Moving Data: General Architecture



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Moving Data: General Architecture

Major functional components:

- **DBMS_DATAPUMP:** Contains the API for high-speed export and import utilities for bulk data and metadata movement
- **Direct Path API (DPAPI):** Oracle Database 11g supports a Direct Path API interface that minimizes data conversion and parsing at both unload and load time.
- **DBMS_METADATA:** Used by worker processes for all metadata unloading and loading. Database object definitions are stored using XML rather than SQL.
- **External Table API:** With the ORACLE_DATAPUMP and ORACLE_LOADER access drivers, you can store data in external tables (that is, in platform-independent files). The SELECT statement reads external tables as though they were stored in an Oracle database.
- **SQL*Loader:** Has been integrated with external tables, providing automatic migration of loader control files to external table access parameters
- **expdp and impdp:** Thin layers that make calls to the DBMS_DATAPUMP package to initiate and monitor Data Pump operations
- **Other clients:** Applications (such as Database Control, replication, transportable tablespaces, and user applications) that benefit from this infrastructure. SQL*Plus may also be used as a client of DBMS_DATAPUMP for simple status queries against ongoing operations.

Oracle Data Pump: Overview

As a server-based facility for high-speed data and metadata movement, Oracle Data Pump:

- Is callable via `DBMS_DATAPUMP`
- Provides the following tools:
 - `expdp`
 - `impdp`
 - Web-based interface
- Provides four data movement methods:
 - Data file copying
 - Direct path
 - External tables
 - Network link support
- Detaches from and reattaches to long-running jobs
- Restarts Data Pump jobs



ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Oracle Data Pump: Overview

Oracle Data Pump enables very high-speed data and metadata loading and unloading of Oracle databases. The Data Pump infrastructure is callable via the `DBMS_DATAPUMP` PL/SQL package. Thus, custom data movement utilities can be built by using Data Pump.

Oracle Database 11g provides the following tools:

- Command-line export and import clients called `expdp` and `impdp`, respectively
- A Web-based export and import interface that is accessible from Database Control

Data Pump automatically decides the data access methods to use; these can be either direct path or external tables. Data Pump uses direct path load and unload when a table's structure allows it and when maximum single-stream performance is desired. However, if there are clustered tables, referential integrity constraints, encrypted columns, or a number of other items, Data Pump uses external tables rather than direct path to move the data.

The ability to detach from and reattach to long-running jobs without affecting the job itself enables you to monitor jobs from multiple locations while they are running. All stopped Data Pump jobs can be restarted without loss of data as long as the metainformation remains undisturbed. It does not matter whether the job is stopped voluntarily or involuntarily due to a crash.

Oracle Data Pump: Benefits

Data Pump offers many benefits and some new features over earlier data movement tools, such as:

- Fine-grained object and data selection
- Explicit specification of database version
- Parallel execution
- Estimation of export job space consumption
- Network mode in a distributed environment
- Remapping capabilities
- Data sampling and metadata compression
- Compression of data during a Data Pump export
- Security through encryption
- Ability to export XMLType data as CLOBs
- Legacy mode to support old import and export files

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Data Pump: Benefits

The `EXCLUDE`, `INCLUDE`, and `CONTENT` parameters are used for fine-grained object and data selection.

You can specify the database version for objects to be moved (using the `VERSION` parameter) to create a dump file set that is compatible with a previous release of the Oracle database that supports Data Pump.

You can use the `PARALLEL` parameter to specify the maximum number of threads of active execution servers operating on behalf of the export job.

You can estimate how much space an export job would consume (without actually performing the export) by using the `ESTIMATE_ONLY` parameter.

Network mode enables you to export from a remote database directly to a dump file set. This can be done by using a database link to the source system.

During import, you can change the target data file names, schemas, and tablespaces.

In addition you can specify a percentage of data to be sampled and unloaded from the source database when performing a Data Pump export. This can be done by specifying the `SAMPLE` parameter.

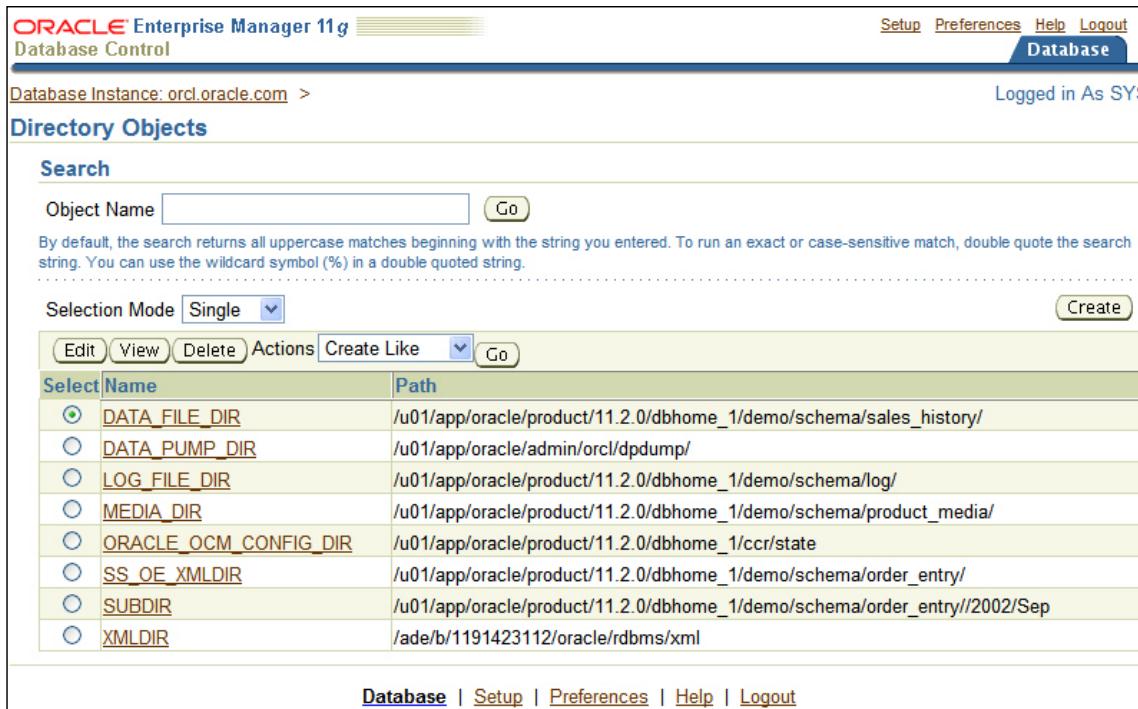
You can use the `COMPRESSION` parameter to indicate whether the metadata should be compressed in the export dump file so that it consumes less disk space. If you compress the metadata, it is automatically uncompressed during import.

Data Pump Enhancements in Oracle Database 11g

In Oracle Database 11g, new features have been added that enable you to:

- Compress both data and metadata, only data, only metadata, or no data during an export
- Specify additional encryption options in the following areas:
 - You can choose to encrypt both data and metadata, only data, only metadata, no data, or only encrypted columns during an export.
 - You can specify a specific encryption algorithm to use during an export.
 - You can specify the type of security to use for performing encryption and decryption during an export. For example, perhaps the dump file set will be imported into a different or remote database and it must remain secure in transit. Or perhaps the dump file set will be imported onsite using the Oracle Encryption Wallet but it may also need to be imported offsite where the Oracle Encryption Wallet is not available.
- Perform table mode exports and imports using the transportable method; specify how partitioned tables should be handled during import operations
- Overwrite existing dump files during an export operation
- Rename tables during an import operation
- Specify that a data load should proceed even if nondeferred constraint violations are encountered (This is valid only for import operations that use the external tables access method.)
- Specify that XMLType columns are to be exported in uncompressed CLOB format regardless of the XMLType storage format that was defined for them
- During an export, specify a remap function that takes as a source the original value of the designated column and returns a remapped value that will replace the original value in the dump file
- Remap data as it is being imported into a new database
- Legacy mode to support the use of original Export (exp) and Import (imp) scripts

Directory Objects for Data Pump



The screenshot shows the Oracle Enterprise Manager 11g Database Control interface. The title bar reads "ORACLE Enterprise Manager 11g Database Control". The top menu bar includes "Setup", "Preferences", "Help", "Logout", and a "Database" button. The sub-menu "Database" is currently selected. The URL in the address bar is "Database Instance: orcl.oracle.com >". The page title is "Directory Objects". A "Search" section contains a "Object Name" input field and a "Go" button. Below it is a note about search behavior. A "Selection Mode" dropdown is set to "Single". A toolbar below the table includes "Edit", "View", "Delete", "Actions", "Create Like", and a "Go" button. The main content is a table with two columns: "Select" and "Name" (which also contains the "Path"). The table lists the following entries:

Select	Name	Path
<input checked="" type="radio"/>	DATA_FILE_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/sales_history/
<input type="radio"/>	DATA_PUMP_DIR	/u01/app/oracle/admin/orcl/dpdump/
<input type="radio"/>	LOG_FILE_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/log/
<input type="radio"/>	MEDIA_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/product_media/
<input type="radio"/>	ORACLE_OCM_CONFIG_DIR	/u01/app/oracle/product/11.2.0/dbhome_1/ocr/state
<input type="radio"/>	SS_OE_XMLDIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/order_entry/
<input type="radio"/>	SUBDIR	/u01/app/oracle/product/11.2.0/dbhome_1/demo/schema/order_entry//2002/Sep
<input type="radio"/>	XMLDIR	/ade/b/1191423112/oracle/rdbms/xml

At the bottom of the page are links for "Database", "Setup", "Preferences", "Help", and "Logout".

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Directory Objects for Data Pump

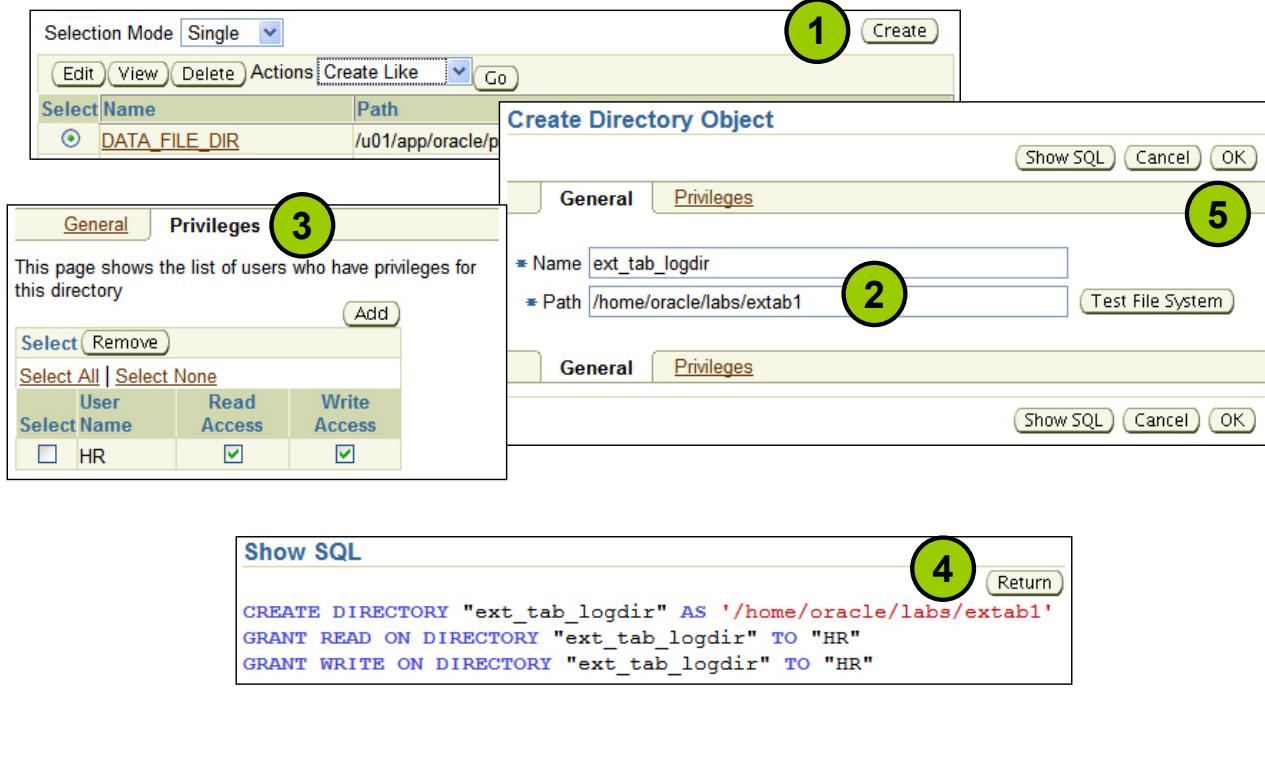
Directory objects are logical structures that represent a physical directory on the server's file system. They contain the location of a specific operating system directory. This directory object name can be used in Enterprise Manager so that you do not need to hard-code directory path specifications. You thus get greater file management flexibility. Directory objects are owned by the `SYS` user. Directory names are unique across the database because all the directories are located in a single name space (that is, `SYS`).

Directory objects are required when you specify file locations for Data Pump because it accesses files on the server rather than on the client.

In Enterprise Manager, select Schema > Database Objects > Directory Objects.

To edit or delete a directory object, select the object and click the appropriate button.

Creating Directory Objects



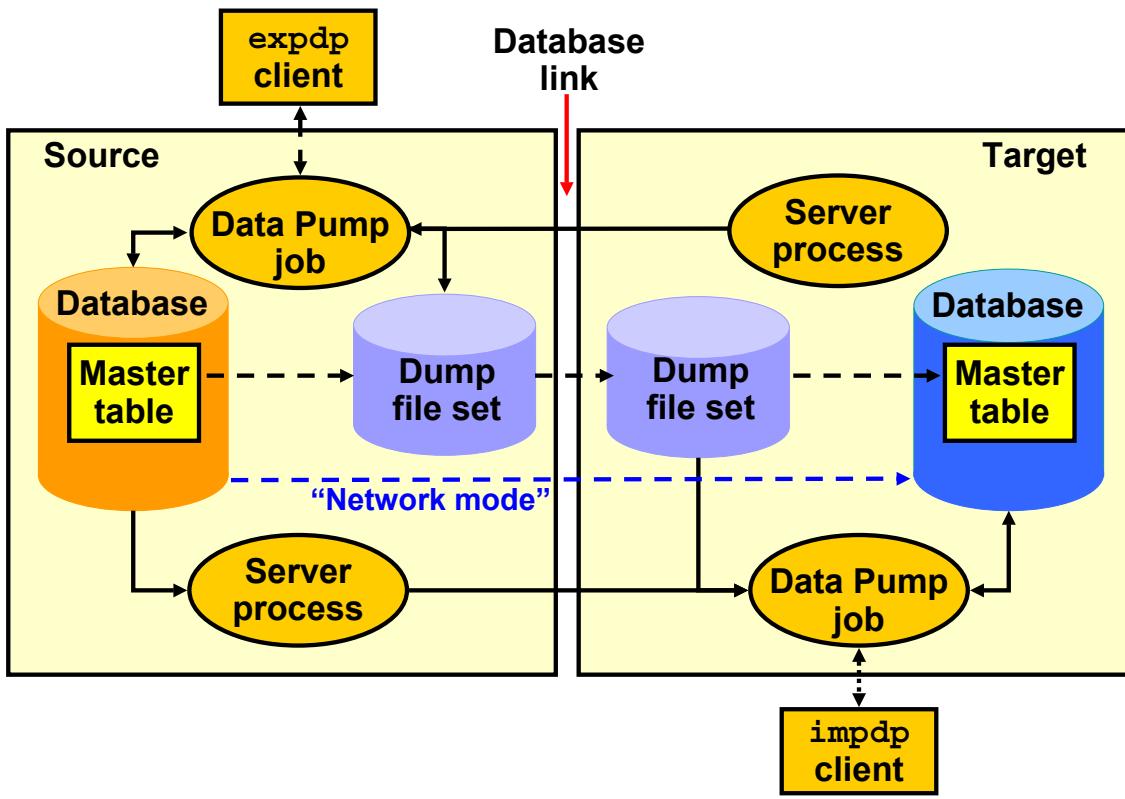
ORACLE

Copyright © 2009, Oracle. All rights reserved.

Creating Directory Objects

1. On the Directory Objects page, click the Create button.
2. Enter the name of the directory object and the OS path to which it maps. OS directories should be created before they are used. You can test this by clicking the Test File System button. For the test, provide the host login credentials (that is, the OS user who has privileges on this OS directory).
3. Permissions for directory objects are not the same as OS permissions on the physical directory on the server file system. You can manage user privileges on individual directory objects. This increases the level of security and gives you granular control over these objects. On the Privileges page, click Add to select the user to which you give read or write privileges (or both).
4. Click Show SQL to view the underlying statements. Click Return when finished.
5. Click OK to create the object.

Data Pump Export and Import Clients: Overview



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Export and Import Clients: Overview

Data Pump Export is a utility for unloading data and metadata into a set of operating system files called *dump file sets*. Data Pump Import is used to load metadata and data stored in an export dump file set into a target system.

The Data Pump API accesses its files on the server rather than on the client.

These utilities can also be used to export from a remote database directly to a dump file set, or to load the target database directly from a source database with no intervening files. This is known as *network mode*. This mode is particularly useful to export data from a read-only source database.

At the center of every Data Pump operation is the master table (MT), which is a table created in the schema of the user running the Data Pump job. The MT maintains all aspects of the job. The MT is built during a file-based export job and is written to the dump file set as the last step. Conversely, loading the MT into the current user's schema is the first step of a file-based import operation and is used to sequence the creation of all objects imported.

Note: The MT is the key to Data Pump's restart capability in the event of a planned or unplanned stopping of the job. The MT is dropped when the Data Pump job finishes normally.

Data Pump Utility: Interfaces and Modes

- Data Pump Export and Import interfaces:
 - Command line
 - Parameter file
 - Interactive command line
 - Enterprise Manager
- Data Pump Export and Import modes:
 - Full
 - Schema
 - Table
 - Tablespace
 - Transportable tablespace



ORACLE

Copyright © 2009, Oracle. All rights reserved.

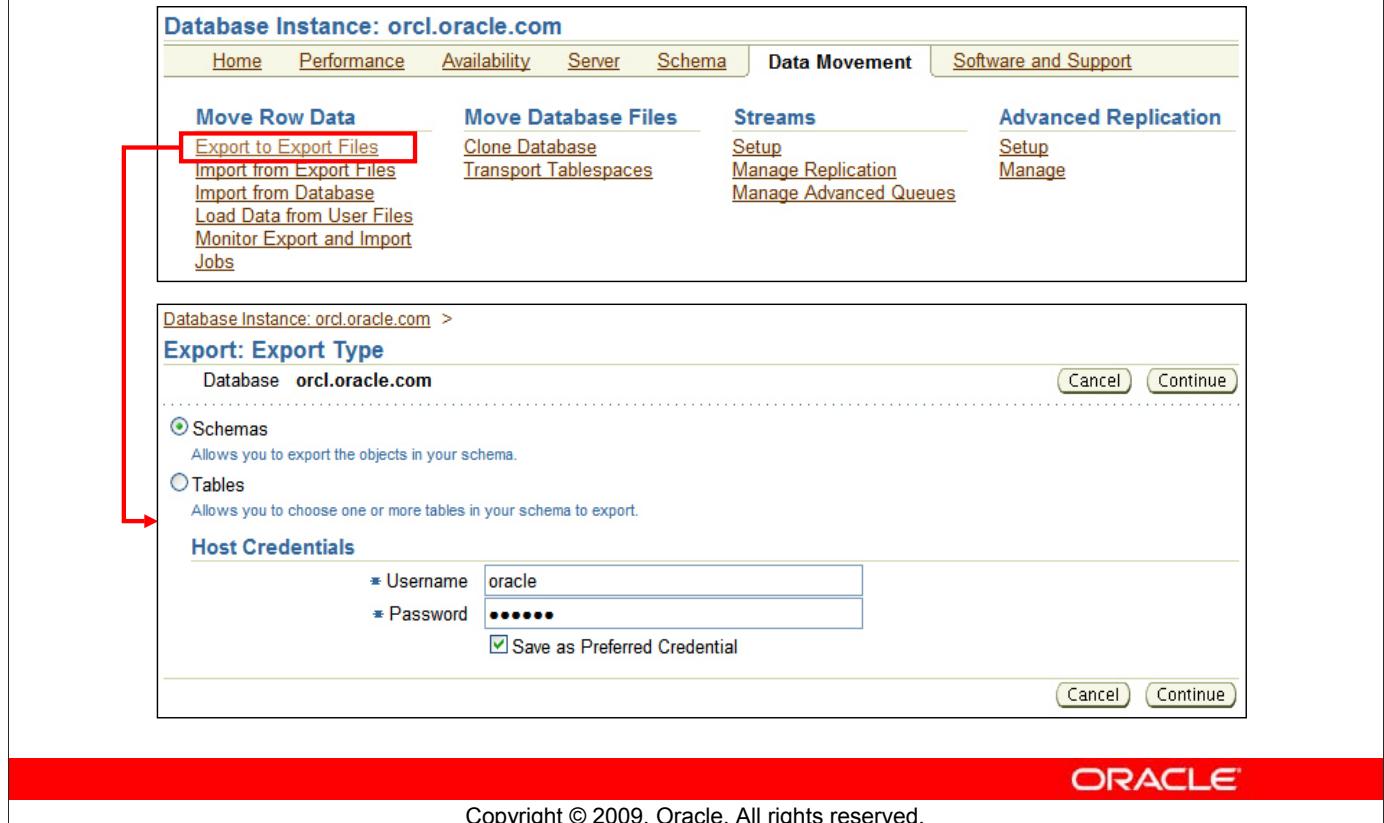
Data Pump Utility: Interfaces and Modes

You can interact with Data Pump Export and Import by using one of the following interfaces:

- **Command line interface:** Enables you to specify most of the export parameters directly on the command line
- **Parameter file interface:** Enables you to specify all command line parameters in a parameter file. The only exception is the PARFILE parameter.
- **Interactive-command interface:** Stops logging to the terminal and displays the export or import prompts, where you can enter various commands. This mode is enabled by pressing [Ctrl] + [C] during an export operation that is started with the command line interface or the parameter file interface. Interactive-command mode is also enabled when you attach to an executing or stopped job.
- **Web interface:** On the Database Control home page, click the Data Movement tab, and then select one of the following links from the Move Row Data region: Export to Export Files, Import from Export Files, or Import from Database.

Data Pump Export and Import provide different modes for unloading and loading different portions of the database. The mode is specified on the command line by using the appropriate parameter. The available modes are listed in the slide and are the same as in the original export and import utilities.

Data Pump Export using Database Control



The screenshot shows the Oracle Enterprise Manager Database Control interface. The main page has a navigation bar with tabs: Home, Performance, Availability, Server, Schema, Data Movement (which is selected and highlighted in blue), and Software and Support. Under the Data Movement tab, there are four main sections: Move Row Data, Move Database Files, Streams, and Advanced Replication. The 'Move Row Data' section contains links for Export to Export Files, Import from Export Files, Import from Database, Load Data from User Files, Monitor Export and Import, and Jobs. A red box highlights the 'Export to Export Files' link. A red arrow points from this link to the 'Export: Export Type' dialog box. The dialog box shows the database instance as 'orcl.oracle.com'. It has two radio button options: 'Schemas' (selected) and 'Tables'. Below these are 'Host Credentials' fields for 'Username' (oracle) and 'Password' (*****). A 'Save as Preferred Credential' checkbox is checked. At the bottom of the dialog are 'Cancel' and 'Continue' buttons. The Oracle logo and copyright notice 'Copyright © 2009, Oracle. All rights reserved.' are at the bottom of the page.

Data Pump Export Using Database Control

Enterprise Manager Database control provides a wizard to guide you through the process of performing a Data Pump export and import procedure. The example above shows a Data Pump export.

From the Database Instance home page, click the Data Movement tab, and locate the Data Pump export and import options under the "Move Row Data" section of the web page. Click "Export to Export Files" to begin a Data Pump export session.

The next window to appear is the selection of export type. If a privileged user is connected to the database instance, then the export types include the following:

- Database
- Schemas
- Tables
- Tablespace

If a non-administrative account is used, then the export type list is limited to the following:

- Schemas
- Tables

Click Continue to proceed with the export.

Data Pump Export Example: Basic Options

Options Files Schedule Review

Export: Options

Database **orcl.oracle.com** **Cancel** **Finish** **Step 1 of 4** **Next**

Maximum Number of Threads in Export Job This option allows you to make tradeoffs between resource consumption and elapsed time. Parallelism is only available in Enterprise Edition.

Estimate Disk Space
Calculates an estimate of how much disk space the export job will consume (in bytes). The estimate is for table row data only and does not include metadata.

Blocks
Estimate will be calculated by multiplying the number of database blocks used by the target objects times the appropriate block sizes. This method will provide the quickest rough estimate.

Statistics
Estimate will be calculated using per-table statistics. This method will provide the most accuracy if all target tables have been recently analyzed.

Estimate Disk Space Now
Calculate the estimate of space that will be consumed without actually performing the export operation. This may take a few minutes.

Optional File

Generate Log File
Directory Object **DATA_PUMP_DIR**
Log File **EXPDAT.LOG**

Show Advanced Options

Cancel **Finish** **Step 1 of 4** **Next**

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Export Example: Basic Options

The Options page displays configurable options for the Data Pump export job. The "Maximum Number of Threads in Export Job" entry corresponds to the `PARALLEL` parameter on the command line. The value you specify should be less than, or equal to, the number of files in the dump file set. This option determines the number of parallel I/O server processes that are used, but the master control process and worker processes acting as query coordinators in parallel query operations do not count toward this total.

An estimation of how much disk space the export job will consume (in bytes) will be printed on the standard output and log file. The estimation can be based on block counts multiplied by block sizes or based on recent statistics analysis. This option corresponds to the `ESTIMATE` command line parameter.

An optional log file can be specified for the export job for all messages regarding work in progress, work completed, and errors encountered. The user performing the export will need write permissions on the directory object specified for the log file. If the log file being specified already exists, it will be overwritten. This option corresponds to the `LOGFILE` parameter on the command line.

Data Pump Export Example: Advanced Options

Content

What to Export from the Source Database All
Export both metadata and data
 Data Only
Export only table row data
 Metadata Only
Export only database object definitions

Export Content Include All Objects
 Include Only Objects Specified Below
 Exclude Only Objects Specified Below

Objects to Include or Exclude

Select Object Type Object Name Expression

No items found

Add Another Row

Object Name Expression example: "IN('EMP','DEPT")" or, to include every object except those of a particular type not beginning with PRO, select EXCLUDE with an expression of "NOT LIKE 'PRO%"

Flashback

Export read-consistent view of data
 As the specified System Change Number (SCN)
SCN
 As the SCN which most closely matches the specified time
Date
Time AM PM

Query

Specify SELECT statement predicate clauses to be applied to tables being exported. If a Table Name is not supplied for a particular Predicate Clause, the Predicate Clause is applied to (and must make sense for) all tables being exported.

Select Predicate Clause Table Name

No items found

Add

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Export Example: Advanced Options

Advanced options can be displayed by clicking the "Show Advanced Options" link.

The "Content" section of the page enables you to filter what export unloads: data only, metadata only, or both. This option corresponds to the CONTENT command line parameter. The "Content" section also allows for the INCLUDE and EXCLUDE command line parameters to be specified.

The EXCLUDE parameter enables any database object type to be excluded from an export or import operation. The optional name qualifier enables you to have finer selectivity within each object type that is specified, as in these examples:

```
EXCLUDE=VIEW  
EXCLUDE=PACKAGE  
EXCLUDE=INDEX : "LIKE 'EMP%' "
```

The INCLUDE parameter includes only the specified object types and objects in an operation.

Syntax: INCLUDE = object_type [: "name_expr"]

Flashback query can be used when data is being exported.

The QUERY parameter operates in a similar manner as the original export utility, with two significant enhancements: It can be qualified with a table name so that it applies to only that table, and it can be used during import as well. Here is an example:

```
QUERY=hr.employees : "WHERE department_id in (10,20)
```

Data Pump Export Example: Files

Select Directory Object	File Name	Maximum File Size (MB)
DATA_PUMP_DIR	HR_SCHEMA.DMP	

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Export Example: Files

The DUMPFILE parameter specifies the names and (optionally) directories of disk-based dump files. Multiple file specifications may be provided as a comma-separated list or in separate DUMPFILE parameter specifications. File names may contain the substitution variable %U, which implies that multiple files may be generated. %U is expanded in the resulting file names into a two-character, fixed-width, monotonically increasing integer starting at 01. If no DUMPFILE is specified, expdat.dmp is used by default. Created dump files are autoextensible by default.

If FILESIZE is specified, each file is FILESIZE bytes in size and nonextensible. If more dump space is required and a template with %U has been supplied, a new file is automatically created with FILESIZE bytes; otherwise, the client receives a message to add a new file.

If a template with %U is specified, the number of files that are initially created is equal to the PARALLEL parameter.

Preexisting files that match the resulting file names are not overwritten by default. Instead, they result in an error and cause the job to be aborted. If you want the files to be overwritten then set REUSE_DUMPFILES=Y. Click Next to proceed with the export.

Note: If multiple dump file templates are provided, they are used to generate dump files in a circular fashion.

Data Pump Export Example: Files (continued)

- You do not need to create a directory object manually before using Data Pump Export. A default directory object named DATA_PUMP_DIR is created for every database, whether newly created or upgraded by a script on UNIX or Windows platforms. Access to the DATA_PUMP_DIR directory is granted automatically to the EXP_FULL_DATABASE and IMP_FULL_DATABASE roles. The DATA_PUMP_DIR directory is created in one of the following locations:
 - <ORACLE_BASE>/admin/DB_UNIQUE_NAME/dpdump
 - <ORACLE_HOME>/admin/DB_UNIQUE_NAME/dpdump

The exact directory path specification for DATA_PUMP_DIR varies depending on the value of the ORACLE_BASE and ORACLE_HOME system environment variables and on the existence of the DATA_PUMP_DIR subdirectory. If ORACLE_BASE is defined on the target system, that value is used. Otherwise, the value of ORACLE_HOME is used. If the DATA_PUMP_DIR subdirectory is for some reason not found, the following default path is used:

ORACLE_HOME/rdbms/log

Note: In all cases, you must have the appropriate access privileges to the directory object for the attempted operation. For export, you need write access for all files; for import, you need read access for dump files and write access for log files and SQL files.

Data Pump Export Example: Schedule

Options Files **Schedule** Review

Export: Schedule

Database **orcl.oracle.com** Cancel Back Step 3 of 4 Next

Specify a name and description for the export job. Specify a date to start the job.

Job Parameters

Job Name **Export_HR_Schema**
Description **Full export of HR schema**

Job Schedule

Time Zone **(UTC+00:00) Universal Time**

Start

Immediately
 Later

Date **Jul 11, 2009**
(example: Jul 11, 2009)

Time **5 20 AM**

Repeat

One Time Only
 Interval
Frequency **1 Minutes**

Monthly
 Yearly

Repeat Until

Indefinite
 Custom

Date **Jul 11, 2009**
(example: Jul 11, 2009)

Time **5 30 AM**
(Ignored except when repeating by minutes or hours.)

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Export Example: Schedule

Data Pump jobs (created through this wizard) can be scheduled as repeatable jobs by Enterprise Manager Database Control. If the Job Name is not specified, a system-generated name will be used. Click Next to proceed with the export.

Data Pump Export Example: Review

Export Type **Schemas**
Statistics type **Estimate optimizer statistics when data is imported**
Parallelism **1**
Files to Export **DATA_PUMP_DIR HR_SCHEMA.DMP**
Log File **DATA_PUMP_DIR EXPDAT.LOG**
Job Schedule **Immediately**

PL/SQL

```
declare
  h1  NUMBER;
begin
  h1 := dbms_datapump.open (operation => 'EXPORT', job_mode => 'SCHEMA', job_name =>
'EXPORT_HR_SCHEMA', version => 'COMPATIBLE');
  dbms_datapump.set_parallel(handle => h1, degree => 1);
  dbms_datapump.add_file(handle => h1, filename => 'EXPDAT.LOG', directory => 'DATA_PUMP_DIR',
filetype => 3);
  dbms_datapump.set_parameter(handle => h1, name => 'KEEP_MASTER', value => 0);
  dbms_datapump.metadata_filter(handle => h1, name => 'SCHEMA_EXPR', value => 'IN("HR")');
```

Processing
Submit job is progressing. This may take some time.
This may take several minutes. This page will automatically forward to the next page when done.

Process is in progress.

TIP This operation cannot be cancelled. It will continue even if the browser window is closed.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Export Example: Review

The "Review" page displays a summary of the information entered and allows you to view to PL/SQL procedure syntax that will be used for the export job. Click the "Submit Job" button to continue. Once the job is submitted, it cannot be cancelled and it is safe to close the browser.

Data Pump Import Example: impdp

Data Pump can be invoked on the command line to allow further command line options to be specified.

```
$ impdp hr DIRECTORY=DATA_PUMP_DIR \
DUMPFILE=HR_SCHEMA.DMP \
PARALLEL=1 \
CONTENT=ALL \
TABLES="EMPLOYEES" \
LOGFILE=DATA_PUMP_DIR:import_hr_employees.log \
JOB_NAME=importHR \
TRANSFORM=STORAGE:n
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Import Example: impdp

Data Pump provides command line clients for import and export operations. The example above illustrates a Data Pump import using the impdp utility. When running Data Pump using the command line, more options are provided. For a complete list of options, please consult the documentation manual "*Oracle Database Utilities 11g Release 2 (11.2)*."

Data Pump Import: Transformations

You can remap:

- Data files by using REMAP_DATAFILE
- Tablespaces by using REMAP_TABLESPACE
- Schemas by using REMAP_SCHEMA
- Tables by using REMAP_TABLE
- Data by using REMAP_DATA

```
REMAP_TABLE = 'EMPLOYEES' : 'EMP'
```

ORACLE

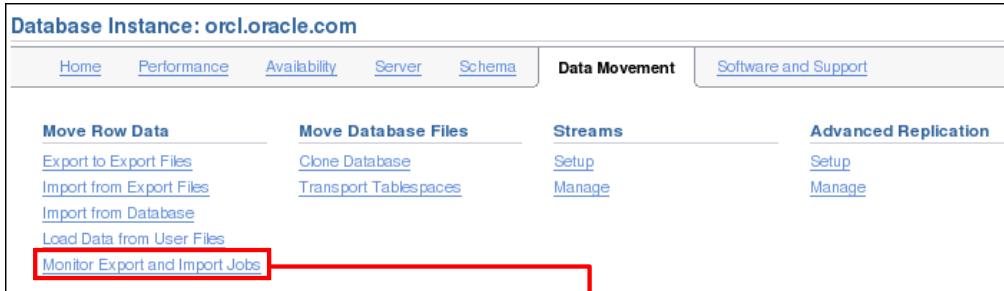
Copyright © 2009, Oracle. All rights reserved.

Data Pump Import: Transformations

Because object metadata is stored as XML in the dump file set, it is easy to apply transformations when DDL is being formed during import. Data Pump Import supports several transformations:

- REMAP_DATAFILE is useful when moving databases across platforms that have different file-system semantics.
- REMAP_TABLESPACE enables objects to be moved from one tablespace to another.
- REMAP_SCHEMA provides the old FROMUSER /TOUSER capability to change object ownership.
- REMAP_TABLE provides the ability to rename entire tables.
- REMAP_DATA provides the ability to remap data as it is being inserted.

Using Enterprise Manager to Monitor Data Pump Jobs



Export and Import Jobs

Page Refreshed Sep 1, 2008 12:23:20 AM MDT

In database versions 10g and greater, Enterprise Manager uses data pump jobs to do the actual export and import operations. Although Enterprise Manager exports and imports can also be monitored from their corresponding Job Summary pages, data pump jobs defined outside of Enterprise Manager can only be monitored from here.

Select Data Pump Job	EM Job	Owner	Job Status
<input type="button" value="INVENTORY EXPORT"/> INVENTORY EXPORT	Yes	DBA1	EXECUTING

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Using Enterprise Manager to Monitor Data Pump Jobs

You can use the Enterprise Manager graphical user interface (GUI) to monitor all Data Pump jobs, including those created by using the `expdp` or `impdp` command line interfaces or by using the `DBMS_DATAPUMP` package.

You can view the current status of the job and change the status to EXECUTE, STOP, or SUSPEND.

To access the “Export and Import Jobs” page, click the “Monitor Export and Import Jobs” link in the Move Row Data region on the Maintenance page.

Migration with Data Pump Legacy Mode

- Assistance in transitioning from `imp` and `exp` utilities to `impdp` and `expdp` utilities
- Data Pump in legacy mode:
 1. Encounters unique `imp` or `exp` parameter and enters legacy mode
 2. Attempts to map the old syntax to the new syntax
 3. Displays new syntax
 4. Exits legacy mode

Best practice tip: Oracle strongly recommends that you view the new syntax and make script changes as time permits.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Migration with Data Pump Legacy Mode

Because of the extensive use of import and export scripts, Oracle Database 11g Release 2 provides migration assistance with Data Pump legacy mode. The Data Pump utilities:

1. Enter legacy mode if a parameter unique to `exp/imp` is present, either on the command line or in a script
2. Map the old parameter to the equivalent `expdp` or `impdp` parameter (if possible)
3. Display the converted command, so that you can view the new syntax and make script changes, as your time permits
4. Exit legacy mode, if a mix of old and new parameters are present (Mixing a Data Pump parameter with an original export or import parameter causes Data Pump to exit without performing the desired task.)

For additional new features, see *Oracle Database Utilities 11g Release 2*.

Data Pump Legacy Mode

The Data Pump export and import utilities:

- Read and write files only in Data Pump format
- Accept `exp` and `imp` utility commands in legacy mode
- Include legacy mode parameters that:
 - Can be identical to the new syntax:
`FILESIZE=integer[B | K | M | G]`
 - Can be similar:
`QUERY= query_clause`
 - Are ignored, when the command is superceded by Data Pump defaults
`BUFFER=integer`
`COMPRESS={y|n}`
`DIRECT={y|n}`
 - Cause an error when old and new syntax is mixed

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Legacy Mode

The Data Pump utilities operate only with files in the Data Pump format. (Files created by the `exp` utility must be read by the `imp` utility.) The Data Pump legacy mode allows you to continue the use of your existing scripts with the Data Pump utilities. But you must use the new Data Pump syntax, if you want to access new features of the database.

When the utilities find unique `exp` or `imp` parameters, they enter legacy mode.

- Identical parameters are used unchanged. Example: The `FILESIZE=integer[B | K | M | G]` parameter specifies the maximum size of a dump file.
- The `QUERY=query_clause` parameter does not cause any conversion, but note that the `expdp` utility handles queries less restrictive than the deprecated `exp` utility. So your query results may be a little different.
- Some parameters are ignored, because they are superceded by new defaults. Examples:
 - The `BUFFER=integer` parameter is ignored, because the `expdp` utility has no conventional path mode.
 - The `COMPRESS={y|n}` parameter is ignored, because the `expdp` utility has no equivalent parameter.
 - The `DIRECT={y|n}` parameter is ignored, because the `expdp` utility determines if the requested export should use direct path or external table mode.
- A mixture of `exp/imp` and Data Pump parameters causes a job to fail.

Data Pump Legacy Mode

- Legacy mode parameters:

- Are mapped to Data Pump parameters, if possible:

```
consistent={y|n} -> FLASHBACK_TIME  
GRANTS=n -> EXCLUDE=CONSTRAINTS  
INDEXES=n -> EXCLUDE=INDEX  
LOG=filename -> LOGFILE=filename  
FILE=filename -> dumpfile=directory-object:filename
```

- Can be similar, but not identical:

```
FEEDBACK=integer -> STATUS
```

- Cause an error when incompatible with new Data Pump:

```
VOLSIZE=integer
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Pump Legacy Mode (continued)

Use the Data Pump legacy mode to transition deprecated scripts to the current version. Several parameters can be mapped to the new syntax:

- Data Pump knows the current time and maps the `CONSISTENT={ y | n }` parameter to the `FLASHBACK_TIME` parameter.
- The `GRANTS=n` parameter is remapped to `EXCLUDE=GRANT`.
- The `INDEXES=n` parameter is remapped to `EXCLUDE=INDEX`.
- The `LOG=filename` parameter is remapped to `LOGFILE=filename`. The contents of the log file (informational and error messages) are in the `expdp` format.
- The `FILE=filename` parameter is remapped to `dumpfile=directory-object:filename`. But if the `expdp` utility cannot find the path to an existing directory object, it aborts.

A parameter can be mapped, but the new functionality is different. The `FEEDBACK=integer` parameter is remapped to `STATUS`. It is not a direct mapping because the status of the export job, not just the number of processed rows, is returned.

A parameter can cause the job to abort, because it is incompatible with Data Pump. The `VOLSIZE=integer` parameter of the `exp` utility specifies the tape volume size. Data Pump does not work with tape drives; Oracle Secure Backup manages tapes.

For a complete mapping of `exp` and `imp` parameters to the Data Pump utilities, see *Oracle Database Utilities 11g Release 2*.

Managing File Locations

- Original `exp` and `imp` utilities: Fully qualified file names
- Data Pump directory object for file locations
 - Default (in prior versions): `DATA_PUMP_DIR` parameter
 - New optional `DATA_PUMP_DIR_<schema-name>` directory object
 - Managed with the `CREATE DIRECTORY` and `GRANT` SQL commands
 - Default location (independent of legacy mode), when:
 - Command line without `DIRECTORY` parameter
 - User without `EXP_FULL_DATABASE` privilege

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Managing File Locations

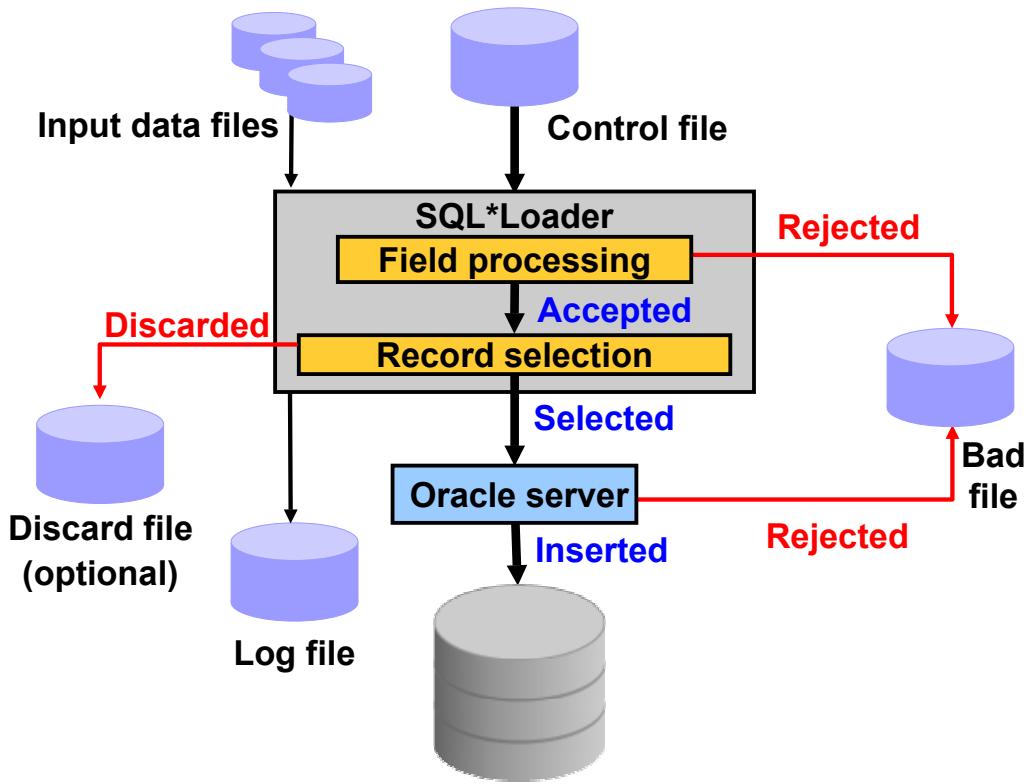
The original and the Data Pump utilities differ in their handling of file locations, because the original utilities were client based (all file names were fully qualified).

The Data Pump utilities are server based. They require that a directory object is used when specifying file locations. The directory object must be accessible to the schema. The default location in prior versions was set by the `DATA_PUMP_DIR` initialization parameter.

This functionality does not introduce new mandatory DBA tasks. It introduces an optional `DATA_PUMP_DIR_<schema-name>` directory object. If you choose to create this directory object with the `CREATE DIRECTORY` and `GRANT` SQL commands, you allow the specified schemas the ability to use Data Pump export or import (which will have an impact on the usual server resources such as CPU, memory usage, and disk usage).

The Data Pump utilities use this directory object when none is specified in the command line and when the user does not have the `EXP_FULL_DATABASE` privilege. This is independent of the legacy mode.

SQL*Loader: Overview



ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL*Loader: Overview

SQL*Loader loads data from external files into tables of an Oracle database. It has a powerful data parsing engine that puts little limitation on the format of the data in the data file.

SQL*Loader uses the following files:

Input data files: SQL*Loader reads data from one or more files (or operating system equivalents of files) that are specified in the control file. From SQL*Loader's perspective, the data in the data file is organized as records. A particular data file can be in fixed record format, variable record format, or stream record format. The record format can be specified in the control file with the `INFILE` parameter. If no record format is specified, the default is stream record format.

Control file: The control file is a text file that is written in a language that SQL*Loader understands. The control file indicates to SQL*Loader where to find the data, how to parse and interpret the data, where to insert the data, and so on. Although not precisely defined, a control file can be said to have three sections.

- The first section contains such session-wide information as the following:
 - Global options, such as the input data file name and records to be skipped
 - `INFILE` clauses to specify where the input data is located
 - Data to be loaded

SQL*Loader: Overview (continued)

- The second section consists of one or more INTO TABLE blocks. Each of these blocks contains information about the table (such as the table name and the columns of the table) into which the data is to be loaded.
- The third section is optional and, if present, contains input data.

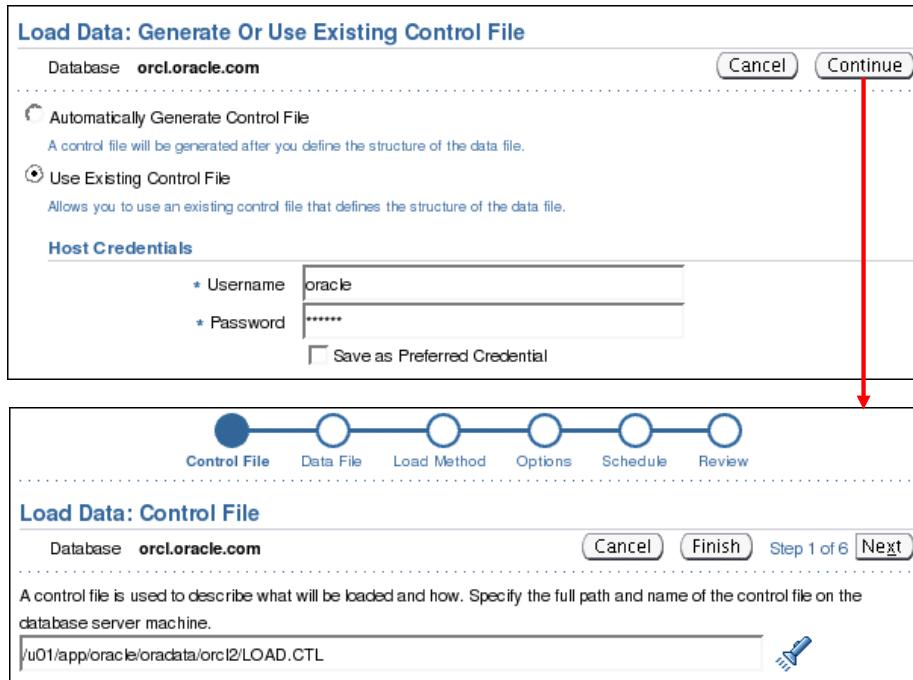
Log file: When SQL*Loader begins execution, it creates a log file. If it cannot create a log file, execution terminates. The log file contains a detailed summary of the load, including a description of any errors that occurred during the load.

Bad file: The bad file contains records that are rejected, either by SQL*Loader or by the Oracle database. Data file records are rejected by SQL*Loader when the input format is invalid. After a data file record is accepted for processing by SQL*Loader, it is sent to the Oracle database for insertion into a table as a row. If the Oracle database determines that the row is valid, the row is inserted into the table. If the row is determined to be invalid, the record is rejected and SQL*Loader puts it in the bad file.

Discard file: This file is created only when it is needed and only if you have specified that a discard file should be enabled. The discard file contains records that are filtered out of the load because they do not match any record-selection criteria specified in the control file.

For more information about SQL*Loader, see the *Oracle Database Utilities* guide.

Loading Data with SQL*Loader



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Loading Data with SQL*Loader

Use the “Load Data from User Files Wizard” to load data from a flat file into an Oracle database.

To display the wizard, select Enterprise Manager Data Movement > Move Row Data > Load Data from User Files.

SQL*Loader Control File

The SQL*Loader control file instructs SQL*Loader about:

- Location of the data to be loaded
- Data format
- Configuration details:
 - Memory management
 - Record rejection
 - Interrupted load handling details
- Data manipulation details



ORACLE

Copyright © 2009, Oracle. All rights reserved.

SQL*Loader Control File

The SQL*Loader control file is a text file that contains data definition language (DDL) instructions. DDL is used to control the following aspects of a SQL*Loader session:

- Where SQL*Loader finds the data to load
- How SQL*Loader expects that data to be formatted
- How SQL*Loader is being configured (including memory management, selection and rejection criteria, interrupted load handling, and so on) as it loads the data
- How SQL*Loader manipulates the data being loaded

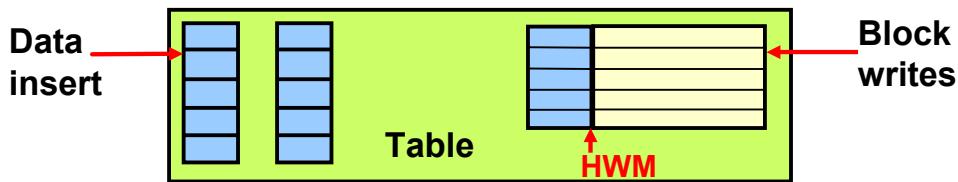
SQL*Loader Control File (continued)

```
1  -- This is a sample control file
2 LOAD DATA
3 INFILE 'SAMPLE.DAT'
4 BADFILE 'sample.bad'
5 DISCARDFILE 'sample.dsc'
6 APPEND
7 INTO TABLE emp
8 WHEN (57) = '.'
9 TRAILING NULLCOLS
10 (hiredate SYSDATE,
    deptno POSITION(1:2) INTEGER EXTERNAL(3)
    NULLIF deptno=BLANKS,
    job POSITION(7:14) CHAR TERMINATED BY WHITESPACE
    NULLIF job=BLANKS "UPPER(:job)",
    mgr POSITION(28:31) INTEGER EXTERNAL
    TERMINATED BY WHITESPACE, NULLIF mgr=BLANKS,
    ename POSITION(34:41) CHAR
    TERMINATED BY WHITESPACE "UPPER(:ename)",
    empno POSITION(45) INTEGER EXTERNAL
    TERMINATED BY WHITESPACE,
    sal POSITION(51) CHAR TERMINATED BY WHITESPACE
    "TO_NUMBER(:sal,'$99,999.99')",
    comm INTEGER EXTERNAL ENCLOSED BY '(' AND '%'
    ":comm * 100"
)
```

The explanation of this sample control file (by line numbers) is as follows:

1. Comments can appear anywhere in the command section of the file, but they must not appear in the data. Precede any comment with two hyphens. All text to the right of the double hyphen is ignored until the end of the line.
2. The LOAD DATA statement indicates to SQL*Loader that this is the beginning of a new data load. If you are continuing a load that has been interrupted in progress, use the CONTINUE LOAD DATA statement.
3. The INFILE keyword specifies the name of a data file containing data that you want to load.
4. The BADFILE keyword specifies the name of a file into which rejected records are placed.
5. The DISCARDFILE keyword specifies the name of a file into which discarded records are placed.
6. The APPEND keyword is one of the options that you can use when loading data into a table that is not empty. To load data into a table that is empty, use the INSERT keyword.
7. The INTO TABLE keyword enables you to identify tables, fields, and data types. It defines the relationship between records in the data file and tables in the database.
8. The WHEN clause specifies one or more field conditions that each record must match before SQL*Loader loads the data. In this example, SQL*Loader loads the record only if the 57th character is a decimal point. That decimal point delimits dollars and cents in the field and causes records to be rejected if SAL has no value.
9. The TRAILING NULLCOLS clause prompts SQL*Loader to treat any relatively positioned columns that are not present in the record as null columns.
10. The remainder of the control file contains the field list, which provides information about column formats in the table that is being loaded.

Loading Methods



Conventional Load	Direct Path Load
Uses COMMIT	Uses data saves (faster operation)
Always generates redo entries	Generates redo only under specific conditions
Enforces all constraints	Enforces only PRIMARY KEY, UNIQUE, and NOT NULL
Fires INSERT triggers	Does not fire INSERT triggers
Can load into clustered tables	Does not load into clusters
Allows other users to modify tables during load operation	Prevents other users from making changes to tables during load operation
Maintains index entries on each insert	Merges new index entries at the end of the load

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Comparing Direct and Conventional Path Loads

Method of Saving Data

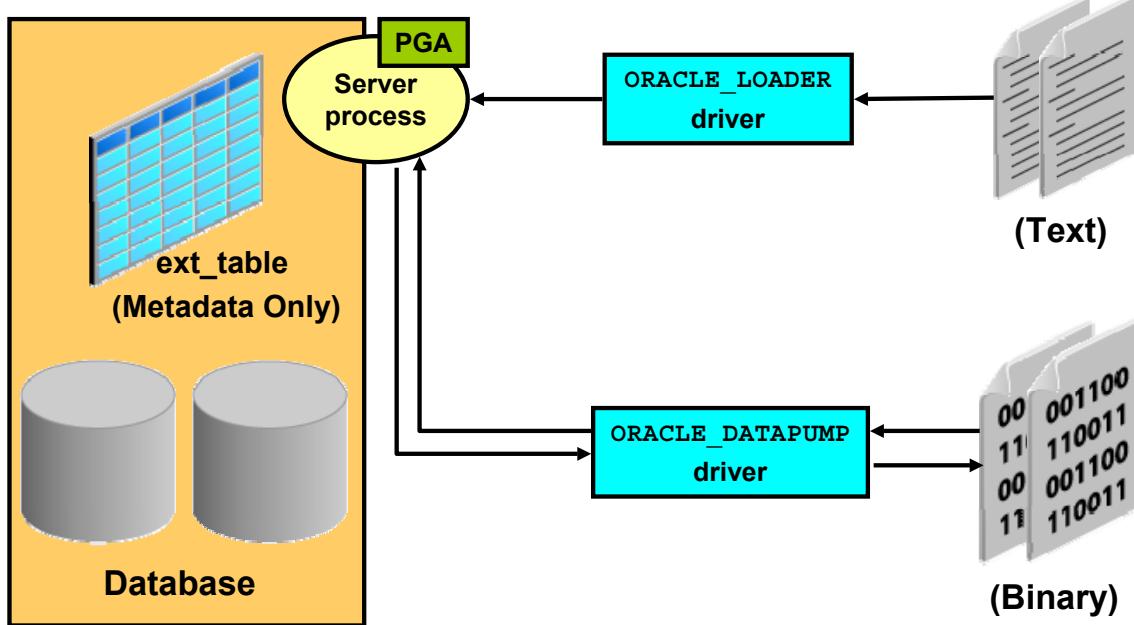
A conventional path load executes SQL `INSERT` statements to populate tables in an Oracle database. A direct path load eliminates much of the Oracle database overhead by formatting Oracle data blocks and writing the data blocks directly to the database files. A direct load does not compete with other users for database resources, so it can usually load data at close to disk speed. Conventional path loads use SQL processing and a database `COMMIT` operation for saving data. The insertion of an array of records is followed by a `COMMIT` operation. Each data load may involve several transactions.

Direct path loads use data saves to write blocks of data to Oracle data files. This is why the direct path loads are faster than the conventional ones. The following features differentiate a data save from `COMMIT`:

- During a data save, only full database blocks are written to the database.
- The blocks are written after the high-water mark (HWM) of the table.
- After a data save, the HWM is moved.
- Internal resources are not released after a data save.
- A data save does not end the transaction.
- Indexes are not updated at each data save.

External Tables

External tables are read-only tables stored as files on the operating system outside of the Oracle database.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

External Tables

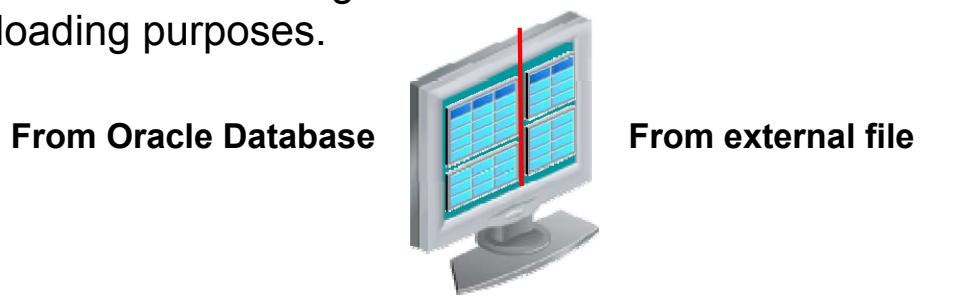
External tables access data in external sources as if it were in a table in the database. You can connect to the database and create metadata for the external table using DDL. The DDL for an external table consist of two parts: one part the describes the Oracle Database column types, and another part that describes the mapping of the external data to the Oracle Database data columns.

An external table does not describe any data that is stored in the database. Nor does it describe how data is stored in the external source. Instead, it describes how the external table layer must present the data to the server. It is the responsibility of the access driver and the external table layer to do the necessary transformations required on the data in the external file so that it matches the external table definition. External tables are read only; therefore, no DML operations are possible, and no index can be created on them.

There are two access driver used with external tables. The ORACLE_LOADER access driver can only be used to read table data from an external table and load it into the database. It uses text files as the data source. The ORACLE_DATAPUMP access driver can both load table data from an external file into the database and also unload data from the database into an external file. It uses binary files as the external files. The binary files have the same format as the files used by the impdp and expdp utilities and can be interchanged with them.

External Table Benefits

- Data can be used directly from the external file or loaded into another database.
- External data can be queried and joined directly in parallel with tables residing in the database, without requiring it to be loaded first.
- The results of a complex query can be unloaded to an external file.
- You can combine generated files from different sources for loading purposes.



ORACLE

Copyright © 2009, Oracle. All rights reserved.

External Table Benefits

The data files created for the external table can be moved and used as the data files for another external table in the same database or different database. External data can be queried and joined directly in parallel to tables residing in the database, without requiring the data to be loaded first. You can choose to have your applications directly access external tables with the `SELECT` command, or you can choose to have data loaded first into a target database.

The results of a complex query can be unloaded to an external file using the `ORACLE_DATAPUMP` access driver.

Data files that are populated by different external tables can all be specified in the `LOCATION` clause of another external table. This provides an easy way of aggregating data from multiple sources. The only restriction is that the metadata for all the external tables must be exactly the same.

Defining an External Tables with ORACLE _ LOADER

```
CREATE TABLE extab_employees
  (employee_id          NUMBER(4),
   first_name           VARCHAR2(20),
   last_name            VARCHAR2(25),
   hire_date             DATE)
ORGANIZATION EXTERNAL
  ( TYPE ORACLE_LOADER DEFAULT DIRECTORY extab_dat_dir
    ACCESS PARAMETERS
      ( records delimited by newline
        badfile extab_bad_dir:'empxt%a_%p.bad'
        logfile extab_log_dir:'empxt%a_%p.log'
        fields terminated by ','
        missing field values are null
      ( employee_id, first_name, last_name,
        hire_date char date_format date mask "dd-mon-yyyy"))
    LOCATION ('empxt1.dat', 'empxt2.dat') )
PARALLEL  REJECT LIMIT UNLIMITED;
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Defining an External Table with ORACLE _ LOADER

The metadata for an external table is created using the SQL language in the database. The ORACLE _ LOADER access driver uses the SQL*Loader syntax to define the external table. This command does not create the external text files.

The example in the slide shows three directory objects (extab_dat_dir, extab_bad_dir, and extab_log_dir) that are created and mapped to existing OS directories to which the user is granted access.

When the table extab_employees is accessed, SQL*Loader functionality is used to load the table, and at that instance the log file and bad file are created.

Best-practice tip: If you have a lot of data to load, enable PARALLEL for the load operation:

```
ALTER SESSION ENABLE PARALLEL DML;
```

External Table Population with ORACLE_DATAPUMP

```
CREATE TABLE ext_emp_query_results
  (first_name, last_name, department_name)
ORGANIZATION EXTERNAL
(
  TYPE ORACLE_DATAPUMP
  DEFAULT DIRECTORY ext_dir
  LOCATION ('emp1.exp','emp2.exp','emp3.exp')
)
PARALLEL
AS
SELECT e.first_name,e.last_name,d.department_name
FROM   employees e, departments d
WHERE  e.department_id = d.department_id AND
       d.department_name in
         ('Marketing', 'Purchasing');
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

External Table Population with ORACLE_DATAPUMP

This example shows you how the external table population operation can help to export a selective set of records resulting from the join of the EMPLOYEES and DEPARTMENTS tables.

Because the external table can be large, you can use a parallel populate operation to unload your data to an external table. As opposed to a parallel query from an external table, the degree of parallelism of a parallel populate operation is constrained by the number of concurrent files that can be written to by the access driver. There is never more than one parallel execution server writing into one file at a particular point in time.

The number of files in the LOCATION clause must match the specified degree of parallelism because each input/output (I/O) server process requires its own file. Any extra files that are specified are ignored. If there are not enough files for the specified degree of parallelism, the degree of parallelization is lowered to match the number of files in the LOCATION clause.

The external table is read-only after it has been populated. The SELECT command can be very complex, allowing specific information to be populated in the external table. The external table, having the same file structure as binary data pump files, can then be migrated to another system, and imported with the impdp utility or read as an external table.

Note: For more information about the ORACLE_DATAPUMP access driver parameters, see the *Oracle Database Utilities 11g Release 2 (11.2)* guide.

Using External Tables

- Querying and external table:

```
SQL> SELECT * FROM extab_employees;
```

- Querying and joining an external table with an internal table

```
SQL> SELECT e.employee_id, e.first_name, e.last_name,
d.department_name FROM departments d, extab_employees e
WHERE d.department_id = e.department_id;
```

- Appending data to an internal table from an external table

```
SQL> INSERT /*+ APPEND */ INTO hr.employees SELECT * FROM
extab_employees;
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using External Tables

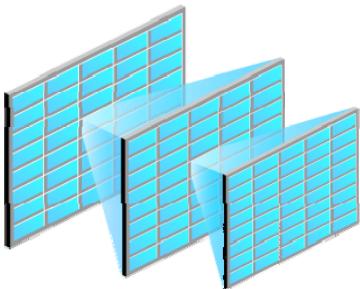
External tables are queried just like internal database tables. The first example illustrates querying an external table EXTAB_EMPLOYEES with only displaying the results. The results are not stored in the database.

The second example shows the joining of an internal table DEPARTMENTS with an external table EXTAB_EMPLOYEES and only displaying the results. The third example on the slide illustrates the direct appending of an internal tables data with the query and load of data from an external table.

Data Dictionary

View information about external tables in:

- [DBA | ALL | USER] _EXTERNAL_TABLES
- [DBA | ALL | USER] _EXTERNAL_LOCATIONS
- [DBA | ALL | USER] _TABLES
- [DBA | ALL | USER] _TAB_COLUMNS
- [DBA | ALL] _DIRECTORIES



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Data Dictionary

The data dictionary views in the slide list the following table information:

[DBA | ALL | USER] _EXTERNAL_TABLES: Specific attributes of external tables in the database

[DBA | ALL | USER] _EXTERNAL_LOCATIONS: Data sources for external tables

[DBA | ALL | USER] _TABLES: Descriptions of the relational tables in the database

[DBA | ALL | USER] _TAB_COLUMNS: Descriptions of the columns of tables, views, and clusters in the database

[DBA | ALL] _DIRECTORIES: Describes the directory objects in the database.

Quiz

Like other database objects, Directory objects are owned by the user that creates them unless another schema is specified during creation.

1. True
2. False



Copyright © 2009, Oracle. All rights reserved.

Answer: 2

Quiz

An index can be created on an external table.

1. True
2. False



Copyright © 2009, Oracle. All rights reserved.

Answer: 2

Summary

In this lesson, you should have learned how to:

- Describe ways to move data
- Create and use directory objects
- Use SQL*Loader to load data from a non-Oracle database (or user files)
- Use external tables to move data via platform-independent files
- Explain the general architecture of Oracle Data Pump
- Use Data Pump Export and Import to move data between Oracle databases

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Practice 17 Overview: Moving Data

This practice covers the following topics:

- Using the Data Pump Export Wizard to select database objects to be exported
- Monitoring a Data Pump Export job
- Using the Data Pump Import Wizard to import tables to your database
- Using the Load Data Wizard to load data into your database
- Loading data by using the command line

ORACLE

Copyright © 2009, Oracle. All rights reserved.

18

Working with Support

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Objectives

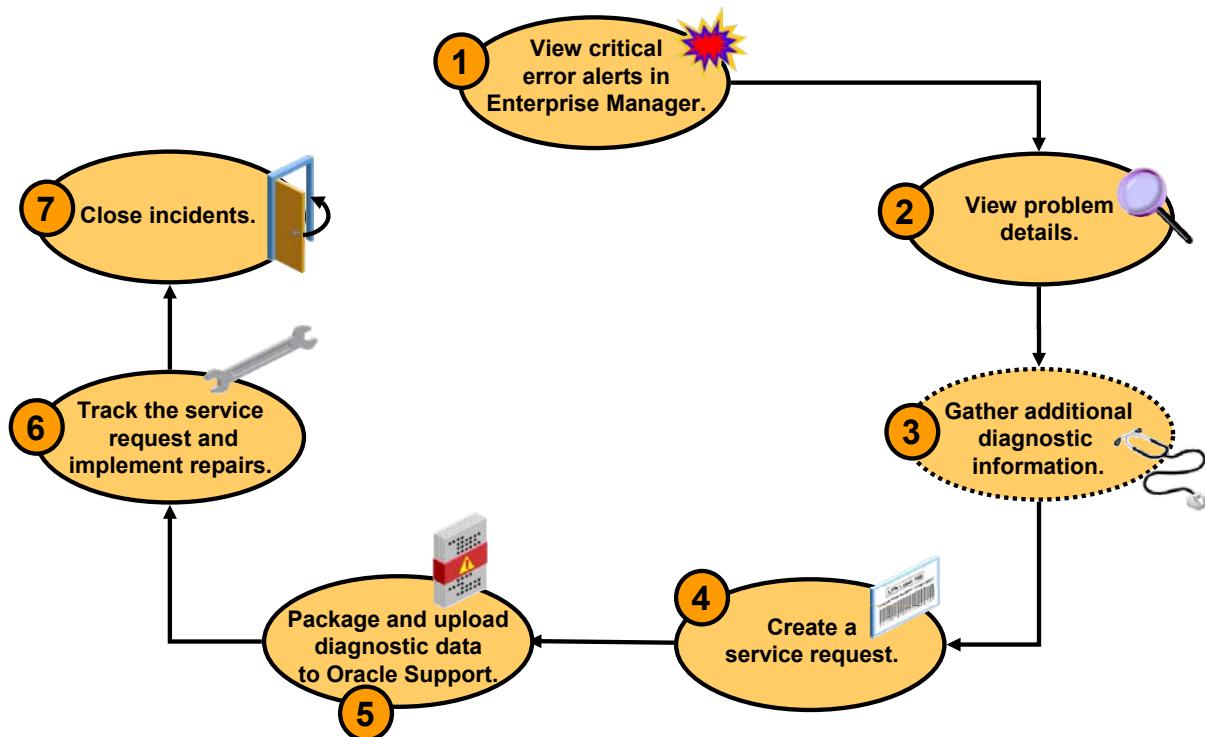
After completing this lesson, you should be able to:

- Use the Enterprise Manager Support Workbench
- Work with My Oracle Support
- Search My Oracle Support
- Log service requests (SR)
- Manage patches
 - Apply a patch
 - Stage a patch



Copyright © 2009, Oracle. All rights reserved.

Using the Support Workbench



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using the Support Workbench

Using the Enterprise Manager Support Workbench, you can investigate, report, and (in some cases) resolve a problem by performing the following general steps:

1. On the Database Home page in Enterprise Manager, review critical error alerts. View the details by selecting an alert.
2. Examine the problem details and view a list of all incidents that were recorded for the problem. Display findings from any health checks that were automatically run.
3. (Optional) Run additional health checks and invoke the SQL Test Case Builder, which gathers all required data related to a SQL problem and packages the information in a way that enables the problem to be reproduced at Oracle Support.
4. Create a service request with My Oracle Support and (optionally) record the service request number with the problem information.
5. Invoke the Incident Packaging Service, which packages all gathered diagnostic data for a problem and (optionally) uploads the data to Oracle Support. You can edit the data to remove sensitive information before uploading.
6. You can maintain an activity log for the service request in the Support Workbench. Run Oracle advisors to help repair SQL failures or corrupted data.
7. Set the status for one, some, or all incidents for the problem to be closed.

Viewing Critical Error Alerts in Enterprise Manager

The screenshot shows the Oracle Enterprise Manager 11g Database Control interface. The main menu bar includes Setup, Preferences, Help, Logout, and Database. The Database Instance is set to database. The Support Workbench page is displayed, showing the following details:

- General:** Status Up, Up Since Apr 2, 2007 12:28:32, Instance Name b, Version 11.1.0.4.0, Host stacq17.us.oracle.com, Listener LISTENER, stacq17.us.oracle.com.
- Diagnostic Summary:** ADDM Findings, Alert Log, No ADDM run available, ARA- errors. A red box highlights the "Active Incidents" link, and a red arrow points to the "Incidents (36)" table.
- Support Workbench:** Problems (5), Checker Findings (36), Packages (4). The "Problems" tab is selected. The table shows the following data:

Select Details	ID	Description	Number of Incidents	Last Incident	Last Comment	Active	Packaged	SR#
<input type="checkbox"/>	1	ORA-1578	36	April 9, 2007 10:00:24 PM PDT	Created package : Id = 5 Name = Pkg_database_ORA_1578_040607125612	Yes	Yes	1234
Incidents (36)								
3953	ORA-1578 [1] [63671]	0	0	0	0	0	0	0
3956	ORA-1578 [1] [63673]	0	0	0	0	0	0	0
3955	ORA-1578 [1] [63673]	0	0	0	0	0	0	0
3954	ORA-1578 [1] [63671]	0	0	0	0	0	0	0
5000	ORA-1578 [1] [63671]	0	0	0	0	0	0	0
5001	ORA-1578 [1] [63673]	0	0	0	0	0	0	0
There are more incidents ...								
►Performance and Critical Error								
Problems (5)	Checker Findings (36)	Packages (4)						

Related Links: Advisor Central, Create User-Reported Problem, Alert Log Contents, Incident Packaging Configuration, Alert Log Errors.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Viewing Critical Error Alerts in Enterprise Manager

You begin the process of investigating problems (critical errors) by reviewing critical error alerts on the Database Home page. To view critical error alerts, access the Database Home page in Enterprise Manager. On the Home page, you can click the Active Incidents link in the Diagnostic Summary section if there are incidents. You can also use the Alerts section and look for critical alerts that are flagged as Incidents.

When you click the Active Incidents link, you access the Support Workbench page, where you can retrieve details about all problems and corresponding incidents. From there, you can also retrieve all Health Monitor checker findings and created packages.

Note: The tasks described in this section are all performed in Enterprise Manager. You can also accomplish all of these tasks with the ADRCI command-line utility. See the *Oracle Database Utilities* guide for more information on the ADRCI utility.

Viewing Problem Details

The screenshot shows the Oracle Support Workbench interface. On the left, a sidebar titled 'Support Workbench' has a 'Problems (5)' section. A red box highlights the 'Select Details' link for incident 3953. The main content area is titled 'Problem Details: ORA 1578'. It includes a 'Summary' section with details like SR# 1234, Bug# --, Active Yes, and Number of Incidents 36. Below this is a 'Last Incident' section. The 'Investigate and Resolve' section contains links for 'Self Service' and 'Oracle Support'. The 'Assess Damage' section includes 'Checker Findings', 'Run Checkers', and 'Database Instance Health'. The 'Diagnose' section includes 'Alert Log', 'Related Problems Across Topology', 'Diagnostic Dumps For Last Incident', and 'Go to Metalink and Research'. The 'Resolve' section includes 'SQL Repair Advisor' and 'Data Recovery Advisor'. At the bottom, the 'Incidents' subpage is shown with a table of incidents. A red box highlights the 'Activity Log' link, which is then expanded to show a table of system-generated comments. The table has columns for 'User', 'Action', 'Description', and 'Timestamp'. The 'Description' column for the first row shows the creation of a package named 'Plg_database_ORA_1578_041007083836'. The 'Timestamp' column shows dates from April 10, 2007, to April 11, 2007. The bottom of the page features a red footer bar with the 'ORACLE' logo and the text 'Copyright © 2009, Oracle. All rights reserved.'

Viewing Problem Details

From the Problems subpage on the Support Workbench page, click the ID of the problem that you want to investigate. This takes you to the corresponding Problem Details page.

On this page, you can see all incidents that are related to your problem. You can associate your problem with a My Oracle Support service request and bug number. In the “Investigate and Resolve” section of the page, you see a Self Service subpage that has direct links to the operations that you can perform for this problem. In the same section, the Oracle Support subpage has direct links to My Oracle Support.

The Activity Log subpage shows you the system-generated operations that have occurred on your problem so far. This subpage enables you to add your own comments while investigating the problem.

On the Incidents subpage, you can click a related incident ID to access the corresponding Incident Details page.

Viewing Incident Details: Dump Files

Contents: b_ora_15201_i3953.trc

File	/ade/aima_emdbsa_b/oracle/log/diag/rdbms/b/b/incident/incdir_3953/b_ora_15201_i3953.trc	
Modified	April 2, 2007 12:27:40 PM PDT	
Size	3.28 MB	

Summary

Problem Key	ORA-1578
[1]	
[63671]	
Status	Ready
Active	No
Timestamp	April 2, 2007 12:27:3
Impact	Unknown
ECID	Unknown
Data Dumped	No
Source	System Generated
Correlation Keys	SID = 129.232, ProcID = 16777216, Client ProcID = oracle@stacg17 (TNS V1-V3).15201_3083220672
Purge Date	April 22, 2007 5:30:24 AM PDT (Purging Enabled) (Disable Purging)

Trace Map
A Trace Map provides a table of contents for a dump file.
TIP Select a section to see its detailed trace records below.

Details

Expand All | Collapse All

Error Stack:
► incident_dump:===== Dump for incident 3953 (ORA 1578) =====
ORA-01578: ORACLE data block corrupted (file # 1, block # 63671)
ORA-01110: data file 1: '/ade/aima_emdbsa_b/oracle/dbs/t_db1.f'

Dump Files [Checker Findings](#) [Additional Diagnostics](#)

File Name	Size (MB)	Timestamp	Path	View Contents
b_ora_15201_i3953.trc	3.28	April 2, 2007 12:27:40 PM PDT	/ade/aima_emdbsa_b/oracle/log/diag/rdbms/b/b/incident/incdir_3953	
b_m000_15218_i3953_87.trc	0.15	April 2, 2007 12:27:42 PM PDT	/ade/aima_emdbsa_b/oracle/log/diag/rdbms/b/b/incident/incdir_3953	

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Viewing Incident Details: Dump Files

When you access the Incident Details page, the Dump Files subpage lists all corresponding dump files. You can then click the eyeglass icon for a particular dump file to visualize the file content with its various sections.

Viewing Incident Details: Checker Findings

Incident Details: 3953

Page Refreshed April 10, 2007 8:08:22 AM PDT (Refresh)

Summary

Problem Key: ORA-1578
[1]
[63671]

Status: Ready
Active: No

Timestamp: April 2, 2007 12:27:37 PM PDT
Impact: Unknown
ECID: Unknown

Data Dumped: No
Source: System Generated
Correlation Keys: SID = 123232, ProcId = 39.24
PQ = (16777216, 1175542056), Client ProcId = oracle@stacg17
(TNS V1-V3),15201_3083220672

Purge Date: April 22, 2007 5:30:24 AM PDT (Purging Enabled)
(Disable Purging)

Checker Findings

Search

Description	Damage Translation	Status	Time Detected
		Open	All

Data Corruption
Select findings and click on the "Launch Recovery Advisor" button to repair those findings.

Launch Recovery Advisor

[Select All](#) | [Select None](#) | [Expand All](#) | [Collapse All](#)

Select	Description	Priority	Damage Translation	Incident ID	Status	Time Detected
<input type="checkbox"/>	All Findings					
<input type="checkbox"/>	Datafile 1: '/ade/aimc_emdbsa_b/oracle/dbs/t_db1.f' contains one or more corrupt blocks	High	Some objects in tablespace SYSTEM might be unavailable	3953	Open	April 2, 2007 12:27:41 PM PDT

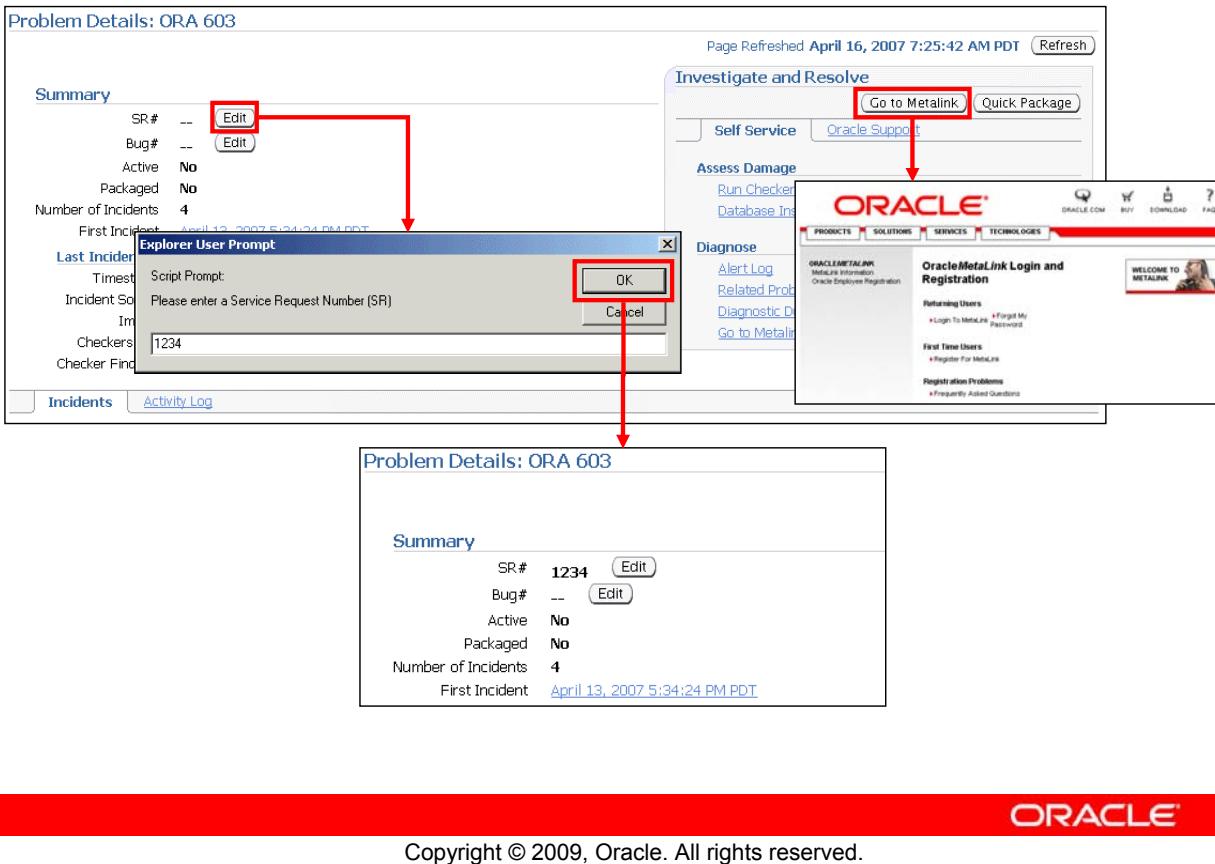
ORACLE

Copyright © 2009, Oracle. All rights reserved.

Viewing Incident Details: Checker Findings

On the Incident Details page, click Checker Findings to view the Checker Findings subpage. This page displays findings from any health checks that were automatically run when the critical error was detected. You will usually have the opportunity to select one or more findings and invoke an advisor to fix the issue.

Creating a Service Request



Creating a Service Request

Before you can package and upload diagnostic information for the problem to Oracle Support, you must create a service request (SR). To create a service request, you first access My Oracle Support. My Oracle Support can be accessed directly from the Problem Details page when you click the “Go to My Oracle Support” button in the “Investigate and Resolve” section of the page. Once on My Oracle Support, log in and create a service request in the usual manner.

When finished, you have the opportunity to enter that service request for your problem. This is entirely optional and is for your reference only.

In the Summary section, click the Edit button that is adjacent to the SR# label. In the window that opens, enter the SR# and then click OK.

Packaging and Uploading Diagnostic Data to Oracle Support

The screenshot shows the Oracle Support Workbench interface. At the top, a summary of a problem (ORA 603) is displayed with fields for SR# (1234), Bug# (--), and Active (No). Below this, the 'Investigate and Resolve' section is shown, with a red box highlighting the 'Quick Package' button. A red arrow points from the 'Quick Package' button to the 'Quick Packaging: Create New Package' dialog. This dialog shows a target database and problems selected (ORA 603). A red box highlights the 'Send to Oracle Support' checkbox. Another red arrow points from this checkbox to the 'Next' button in the wizard. The wizard has four steps: Step 1 of 4 (Create New Package), Step 2 of 4 (View Contents), Step 3 of 4 (View Manifest), and Step 4 of 4 (Schedule). The 'View Contents' step is currently active. It shows the package details: Target database, Problems Selected (ORA 603), Package Name (Pkg_database_ORA_603_041607080712), and Service Request Number (SR#) (1234). A red box highlights the 'Send to Oracle Support' checkbox again. Below this, a table titled 'Incidents to be Packaged' lists several incidents with their IDs, types, and descriptions. The Oracle logo is at the bottom right, and a copyright notice is at the bottom center.

Packaging and Uploading Diagnostic Data to Oracle Support

The Support Workbench provides two methods for creating and uploading an incident package: the Quick Packaging method and the Advanced Packaging method. The example in the slide shows how to use Quick Packaging.

Quick Packaging is a more automated method with a minimum of steps. You select a single problem, provide an incident package name and description, and then schedule the incident package upload, either immediately or at a specified date and time. The Support Workbench automatically places diagnostic data related to the problem into the incident package, finalizes the incident package, creates the ZIP file, and then uploads the file. With this method, you do not have the opportunity to add, edit, or remove incident package files or add other diagnostic data such as SQL test cases.

To package and upload diagnostic data to Oracle Support:

1. On the Problem Details page, in the Investigate and Resolve section, click Quick Package. The Create New Package page of the Quick Packaging wizard appears.
2. Enter a package name and description.
3. Enter the service request number to identify your problem.
4. Click Next, and then proceed with the remaining pages of the Quick Packaging Wizard. Click Submit on the Review page to upload the package.

Tracking the Service Request and Implementing Repairs

Problem Details: ORA 603

Page Refreshed April 16, 2007 8:39:07 AM PDT (Refresh)

Summary

SR #	1234	Edit
Bug#	--	Edit
Active	No	
Packaged	Yes	
Number of Incidents	4	
First Incident	April 13, 2007 5:34:24 PM PDT	
Last Incident		
Timestamp	April 13, 2007 6:40:24 PM PDT	
Incident Source	System Generated	
Impact		
Checkers Run	0	
Checker Findings	0	

Investigate and Resolve

[Go to Metalink](#) [Quick Package](#)

[Self Service](#) [Oracle Support](#)

Collect and Send Diagnostic Data

[Create a Service Request with Metalink](#)
[Record Service Request Number to Problem](#)
[Generate Additional Dumps and Test Cases](#)
[Package the Problem](#)
[View/Send Upload Files](#)

Track and Close

[Check the Service Request Status with Metalink](#)
[Close the problem](#)

[Incidents](#) [Activity Log](#)

Comment [Add Comment](#)

User	Action	Description	Timestamp
SYS	Comment	Set SR : 1234	April 16, 2007 8:34:45 AM PDT
SYS	Comment	Set SR : null	April 16, 2007 8:34:30 AM PDT
SYS	Package	Failed to send upload file to Oracle: packageId = 1 file = /ade/aimc_emdba_b/oracle/stacg17.us.oracle.com_b/sysman/emd/state/Pkg_database_ORA_603_041607080712_COM_1.zip	April 16, 2007 8:14:12 AM PDT
SYS	Package	Created physical file : packageId = 1 file = /ade/aimc_emdba_b/oracle/stacg17.us.oracle.com_b/sysman/emd/state/Pkg_database_ORA_603_041607080712_COM_1.zip	April 16, 2007 8:14:10 AM PDT
SYS	Comment	Created package : Id = 1 Name = Pkg_database_ORA_603_041607080712	April 16, 2007 8:09:30 AM PDT

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Tracking the Service Request and Implementing Repairs

After uploading diagnostic information to Oracle Support, you can perform various activities to track the service request and implement repairs. Among these activities are the following:

- Add an Oracle bug number to the problem information. On the Problem Details page, click the Edit button that is adjacent to the Bug# label. This is for your reference only.
- Add comments to the problem activity log:
 1. Access the Problem Details page for the problem.
 2. Click Activity Log to display the Activity Log subpage.
 3. In the Comment field, enter a comment and then click Add Comment.Your comment is recorded in the activity log.
- Respond to a request by Oracle Support to provide additional diagnostics. Your Oracle Support representative can provide instructions for gathering and uploading additional diagnostics.

Tracking the Service Request and Implementing Repairs

Incident Details: 3953

Page Refreshed April 10, 2007 8:08:22 AM PDT [Refresh](#)

Summary

Problem Key	ORA-1578
[1]	
[63671]	
Status	Ready
Active	No
Timestamp	April 2, 2007 12:27:37 PM PDT
Impact	Unknown
ECID	Unknown
Data Dumped	No
Source	System Generated
Correlation Keys	SID = 129.232, ProcId = 39.24 PQ = (16777216, 1175542056), Client ProcId = oracle@stacg17 (TNS V1-V3).15201_3083220672
Purge Date	April 22, 2007 5:30:24 AM PDT (Purging Enabled) (Disable Purging)

[Dump Files](#) [Checker Findings](#) [Additional Diagnostics](#)

Search

Description	Damage Translation	Status	Time Detected
<input type="text"/>	<input type="text"/>	<input type="button" value="Open"/>	<input type="button" value="All"/>
<input type="button" value="Go"/>			

Data Corruption
Select findings and click on the "Launch Recovery Advisor" button to repair those findings.

[Launch Recovery Advisor](#)

[Select All](#) [Select None](#) [Expand All](#) [Collapse All](#)

Select Description	Priority	Damage Translation	Incident ID	Status	Time Detected
<input type="checkbox"/> All Findings					
<input type="checkbox"/> Datafile 1: '/ade/aime_emdbsa_b/oracle/dbs/t_db1.f' contains one or more corrupt blocks	High	Some objects in tablespace SYSTEM might be unavailable	3953	Open	April 2, 2007 12:27:41 PM PDT

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Tracking the Service Request and Implementing Repairs (continued)

From the Incident Details page, you can run an Oracle advisor to implement repairs. Access the suggested advisor in one of the following places:

- On the Self-Service tab of the "Investigate and Resolve" section of the Problem Details page
- On the Checker Findings subpage of the Incident Details page (as shown in the slide)

The advisors that help you repair critical errors are:

- **Data Recovery Advisor:** Corrupted blocks, corrupted or missing files, and other data failures
- **SQL Repair Advisor:** SQL statement failures

Closing Incidents and Problems

Problem Details: ORA 1578

Page Refreshed July 9, 2007 12:53:11 AM GMT+07:00 Refresh

Summary

SR #	--	Edit
Bug #	--	Edit
Active	Yes	
Packaged	No	
Number of Incidents	1	

Last Incident

Timestamp	July 9, 2007 12:43:48 AM GMT+07:00
Incident Source	System Generated
Impact	
Checkers Run	1
Checker Findings	1

Investigate and Resolve

[Go to Metalink](#) [Quick Package](#)

[Self Service](#) [Oracle Support](#)

Collect and Send Diagnostic Data

[Create a Service Request with Metalink](#)

[Report Service Request Number to](#)

Confirmation

Are you sure you want to close the problem: ORA 1578?
Once the problem is closed, the associated data will be purged after 30 days.

[No](#) [Yes](#)

Track and Close

[Check the Service Request Status with Metalink](#)

[Close the problem](#)

Confirmation

Are you sure you want to close the incident: 30177?
Once the incident is closed, the associated data will be purged after 30 days.

[No](#) [Yes](#)

Incidents [Activity Log](#)

Status	Open Incidents	Data Dumped	Timestamp
View	Close	Yes	July 9, 2007 12:43:48 AM GMT+07:00

[Select All](#) [Select None](#) [Show All Details](#) [Hide All Details](#)

Select Details	ID	Description	Data Dumped	Active	Status	Timestamp
<input checked="" type="checkbox"/>	Show 30177	ORA 1578 [9] [44] [] [] [] [] []	Yes	Yes	Ready	July 9, 2007 12:43:48 AM GMT+07:00

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Closing Incidents and Problems

When a particular incident is no longer of interest, you can close it. By default, closed incidents are not displayed on the Problem Details page. All incidents, whether closed or not, are purged after 30 days. You can disable purging for an incident on the Incident Details page.

To close incidents:

1. Access the Support Workbench home page.
2. Select the desired problem, and then click View.
The Problem Details page appears.
3. Select the incidents to close, and then click Close.
A confirmation page appears.
4. Click Yes on the Confirmation page to close your incident.

Incident Packaging Configuration

The screenshot shows the Oracle Support Workbench interface. On the left, there is a sidebar with 'Support Workbench' and 'Problems (4)'. The main area displays a table of incidents with columns for ID, Description, and Number of Incidents. Below the table are sections for 'Performance and Critical Error' and 'Related Links' (which includes 'Advisor Central', 'Create User-Reported Problem', 'Alert Log Contents', 'Incident Packaging Configuration' (which is highlighted with a red box), and 'Alert Log Errors'). A large red box highlights the 'Edit' and 'OK' buttons in the bottom right corner of the 'Edit Incident Packaging Configuration' dialog box. Another red box highlights the 'Edit' and 'OK' buttons in the bottom right corner of the 'Incident Data Retention' section of the dialog box. A red arrow points from the 'Incident Packaging Configuration' link in the Related Links section to the 'Edit' button in the dialog box.

Incident Packaging Configuration

You can configure retention rules and packaging generation. Access the Incident Packaging configuration page from the Related Links section of the Support Workbench page by clicking the Incident Package Configuration link. Here are the parameters you can change:

- **Incident Metadata Retention Period:** Metadata is information about the data. For incidents, it includes the incident time, ID, size, and problem. Data is the actual content of an incident (such as traces).
- **Cutoff Age for Incident Inclusion:** This value includes incidents for packaging that are in the range to now. When you set the cutoff date to 90, the system includes only those incidents that are within the last 90 days.
- **Leading Incidents Count and Trailing Incidents Count:** For every problem included in a package, the system selects a certain number of incidents from the problem from the beginning (leading) and the end (trailing). For example, if the problem has 30 incidents and the leading incident count is 5 and the trailing incident count is 4, the system includes the first 5 incidents and the last 4 incidents.
- **Correlation Time Proximity:** This parameter is the time interval that defines “happened at the same time.” Correlating incidents (or problems) with certain other incidents or problems helps you answer the question “Which problems seem to have a connection with each other?” One criterion for correlation is time correlation: Find the incidents that happened at the same time as the incidents in a certain problem.

Enterprise Manager Support Workbench for ASM

The screenshot shows the Oracle Enterprise Manager 11g Support Workbench for ASM. At the top, it displays 'ORACLE Enterprise Manager 11g Database Control' and the URL 'Automatic Storage Management +ASM edrsr22p1.us.oracle.com > Support Workbench'. The page is refreshed on June 25, 2009, at 8:5. Below the header, there are three tabs: 'Problems (0)', 'Checker Findings (0)', and 'Packages (0)'. The 'Problems (0)' tab is selected. It shows statistics: 'New Problems in Last 24 Hours 0', 'All Active Problems 0', 'All Problems 0', 'New Incidents in Last 24 Hours 0', 'All Active Incidents 0', and 'All Incidents 0'. A dropdown menu 'View' is set to 'Last 24 Hours'. A search bar is also present. Below the statistics is a table with columns: 'Select', 'Details', 'ID Description', 'Number Of Incidents', 'Last Incident', and 'Last Comment'. The table shows 'No problems found.' A second table with the same structure is shown below, also indicating 'No problems found.' At the bottom, there are 'Related Links' including 'Alert Log Contents', 'Alert Log Errors', 'Create User-Reported Problem', 'Incident Packaging Configuration', and 'Checker Central'. The Oracle logo is at the bottom right, and a copyright notice 'Copyright © 2009, Oracle. All rights reserved.' is at the bottom center.

Enterprise Manager Support Workbench for ASM

Oracle Enterprise Manager has been enhanced to help diagnose and package incidents to Oracle Support Services for Oracle ASM instances.

Oracle Enterprise Manager provides Oracle ASM Support Workbench to monitor Oracle ASM alerts and incidents.

To access Support Workbench for Oracle ASM:

1. Click the Software and Support tab on the database home page.
2. Click Support Workbench in the Support section of the Software and Support page.
3. Click Support Workbench (*ASM_instance_name*) under the Related Links section on the Support Workbench page.

You can view information about current and past problems on the Problems page.

To create a package to send to Oracle Support Services, select an incident and click Package on the Support Workbench Problems page. Support Workbench then guides you through the packaging process.

Working with Oracle Support

- Oracle Support Services (OSS) provides 24×7 solution support.
- Support is delivered in the following ways:
 - My Oracle Support Web site
 - Telephone
 - Oracle Direct Connect (ODC) remote diagnostic tool
- The Customer Support Identifier (CSI) number is used to track the software and support that are licensed to each customer.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Working with Oracle Support

Oracle Support Services (OSS) provides 24×7 solution support to all Oracle customers throughout the world. OSS has support centers around the globe to provide this coverage whenever it is required, 365 days a year.

Support is delivered to Oracle customers through the My Oracle Support Web site, on the telephone, and by using the Oracle Direct Connect (ODC) remote diagnostic tool.

After purchasing Oracle software, customers are provided with a Customer Support Identifier (CSI) number. This number is used to track the software and support licensed to each customer. The CSI number provides access to all the available patches, documentation, and troubleshooting information on My Oracle Support. The CSI number enables customers to log a service request (SR) with OSS.

Note: Service requests were formerly called technical assistance requests (TARs).

My Oracle Support Integration

- Enterprise Manager automatically alerts users to new critical patches.
- The Enterprise Manager patch wizard can be used to select an interim patch.
- You can review the patch's README file from within Enterprise Manager.
- You can download the selected patches from My Oracle Support into the Enterprise Manager patch cache.



Copyright © 2009, Oracle. All rights reserved.

My Oracle Support Integration

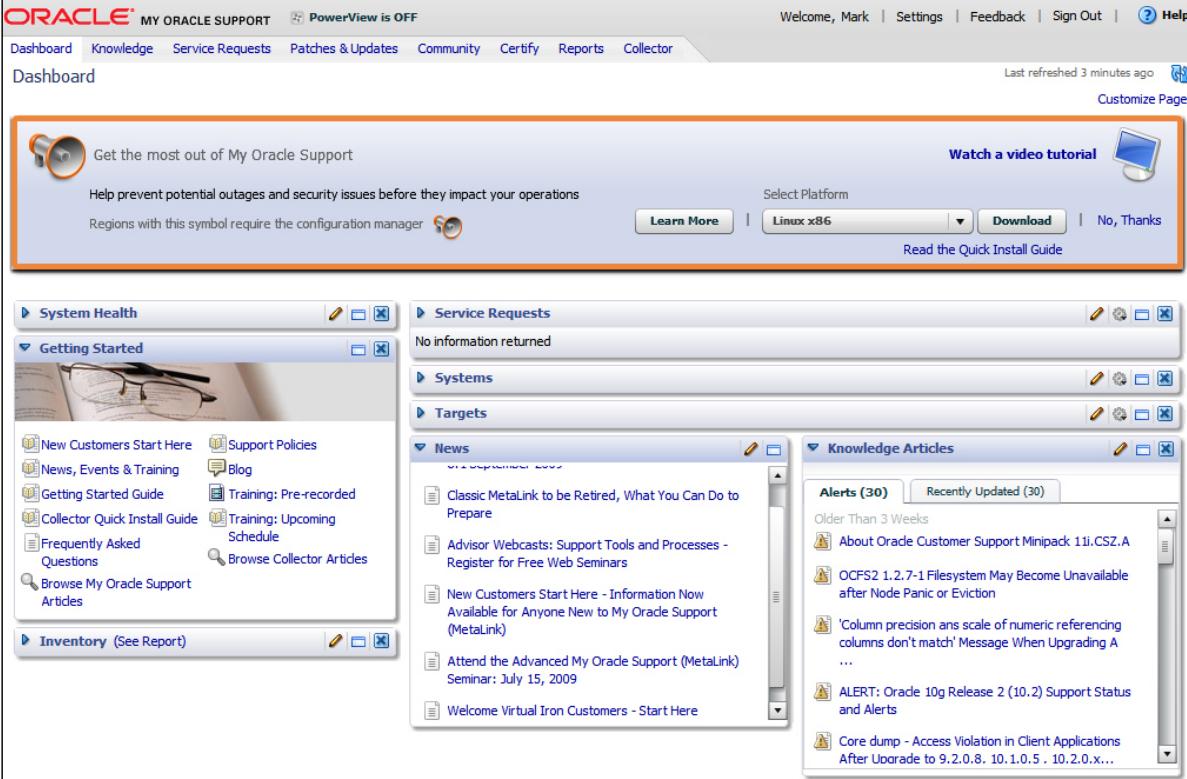
Oracle Enterprise Manager (Enterprise Manager) significantly facilitates software patching with its built-in My Oracle Support integration. Enterprise Manager automatically alerts users to new critical patches and flags all systems that require a specific patch. You can invoke the Enterprise Manager patch wizard to determine what interim patches are available for installation.

Alternatively, you can use the patch wizard to select an interim patch and determine whether any of your systems require that patch. You can review the patch details and README patch notes directly from within Enterprise Manager.

You can use the Enterprise Manager patch wizard to download interim patches from My Oracle Support into the Enterprise Manager patch cache, eliminating the need for repeated downloads. You can stage appropriate patches on the destination system or systems for manual application at a later time. To further automate the patching process, you can also provide a customizable patch application script that is executed on the destination system at a user-defined time by the resident Enterprise Manager agents. As patches are applied to a system, the corresponding Oracle Universal Installer (OUI) inventory is automatically updated to keep track of the systems' correct patch level.

Click Patch in the Deployments region of the Maintenance page to access the patch wizard.

Using My Oracle Support



The screenshot shows the Oracle My Oracle Support interface. At the top, there's a banner with a megaphone icon and text: 'Get the most out of My Oracle Support', 'Help prevent potential outages and security issues before they impact your operations', and 'Regions with this symbol require the configuration manager'. Below the banner, there are tabs for 'System Health', 'Service Requests', 'Systems', 'Targets', 'News', and 'Knowledge Articles'. The 'News' section is expanded, showing a list of articles. The 'Knowledge Articles' section is also expanded, showing a list of alerts. The bottom of the page has a red footer with the Oracle logo and the text 'Copyright © 2009, Oracle. All rights reserved.'

Using My Oracle Support

To register for My Oracle Support, go to <http://myoraclesupport.oracle.com/> and select the link Register here. At the prompt, enter your CSI number and answer a few basic questions. After registering, you are ready to use My Oracle Support. Note that each CSI number has an administrator designated by the customer who controls new-user access to My Oracle Support. Customers must designate this individual, and then new users must work with this individual to create new accounts and grant appropriate My Oracle Support access.

My Oracle Support has a variety of tools and methods available for researching problems.

Searching for answers on My Oracle Support through the standard and advanced search engines is relatively straightforward. A common problem is that too many results are returned. The following are some simple steps that can improve the quality and relevance of search results:

- Use full and exact error text when performing your search. For example, ORA-1400 : mandatory (NOT NULL) column returns more relevant answers than ORA-1400.
- When researching errors in Oracle E-Business Suite, enter the name of the code as part of the search criteria. For example, APXINWKB ORA-1400 : mandatory (NOT NULL) column returns fewer and better results than if you supply only the error message.

Using My Oracle Support (continued)

You can use the Knowledge tab to access the Knowledge Browser if you prefer a drill-down method of searching for information rather than searching by keyword. The Knowledge Browser provides easy-to-use access to OSS's most frequently used technical content.

The Knowledge Browser is organized to provide up-to-date information at your fingertips:

- Recent announcements and information in the *Featured News and Articles* section
- Information by product category
- Case studies
- Tools and training
- Online documentation
- Electronic technical reference manuals (eTRMs)
- Oracle Integration Repository
- Customer Knowledge Exchange

My Oracle Support Forums (Forums) enable you to interact with other Oracle customers to share ideas and discuss Oracle products. You can use My Oracle Support Forums to find out how other customers perform complex tasks or meet various business requirements with Oracle products.

You should not use Forums as a substitute for logging an SR.

Customers can use the patch engine to search for patches by using a variety of methods. The following are the most common patch searches:

- **Patch Number:** If you know the patch number, you can enter it.
- **Latest Consolidated Patch:** You can use this when upgrading to determine the latest patches for the products you are using.
- **Includes File:** When a problem is encountered in a specific piece of code, a patch is often available to fix the issue. For this reason, support representatives often recommend that customers apply a patch to update code to the most current version available for the release. You can find and apply the latest versions of Oracle software by identifying the name and version of the code and then using the patch search utility to find out whether a more current version of the code is available.

Note: For detailed information about performing these searches, refer to My Oracle Support Technical Note 166650.1 (“Working Effectively with Global Customer Support”).

You can use the BUGs link to search the BUG database when researching issues. A variety of methods are available for searching the BUG database.

Researching an Issue

To research an issue on My Oracle Support, perform the following steps:

1. Perform a keyword search.
2. Review the documentation.
3. Use the self-service toolkits.
4. Use the automated diagnostic tests and business flows.
5. Search for applicable patches.
6. Log a service request (SR).

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Researching an Issue

My Oracle Support provides several resources that can be used to research an issue. The following steps outline basic troubleshooting techniques that use My Oracle Support resources:

1. **Keyword search:** Most issues can be resolved quickly and easily by using the keyword search utility on My Oracle Support. Effective searches can provide much information about a specific problem and its solutions.
2. **Documentation:** If keyword searching fails to yield a solution, you should review the documentation to ensure that setup problems are not the root cause. Setup issues account for more than one-third of all service requests; it is always good to review setups early in the troubleshooting process. Documentation consists of user guides and implementation manuals published in PDF format as well as product README files and installation notes published in HTML. Both of these document types are available on My Oracle Support and can be accessed through the self-service toolkits for each product.

Researching an Issue (continued)

3. **Self-service toolkits:** Self-service toolkits (SSTKs) provide a wealth of information about each product. In most cases, they contain FAQs, patch listings, and other helpful information that can assist you in researching and troubleshooting the issues that you are facing. Because SSTKs contain the most frequently used content about each product, you should reference them periodically to identify known issues before they cause problems within your environment.
4. **Diagnostics and flows:** Many recent innovations in Oracle Support Services have been in the area of automated diagnostic tests and business flows. Tests and flows have been created for you to check the setup of your system or gather information about a problem. In the case of diagnostic tests, this can be done by running a Java or SQL script. The output of these tests can help you in resolving issues and can also help Oracle Support Services identify the cause of your problem if it becomes necessary to log a service request.
5. **Patches and BUGs:** There are times when BUGs are found in Oracle products, and patches are required to correct the problem. When troubleshooting a problem, you should review your system to see whether patches are available to provide you with a more recent release of the product. With the patch search tool, you can search for patches that contain specific files. Searching for the latest code and patching your environment to the most recent version improves the troubleshooting process by eliminating existing BUGs that could be possible candidates for the problem. You should also leverage the BUG search engine to see whether a BUG has been logged for your issue but not yet fixed.
6. **Logging a service request (SR):** When all self-service options fail, it may become necessary to engage a support representative to assist in resolving your issue.

Logging Service Requests

- Log an SR by clicking the Service Request tab on the My Oracle Support home page.
- My Oracle Support performs searches based on the CSI number and SR profile.
- Provide the following information when logging an SR:
 - Explanation of the issue, including error messages
 - Steps taken to troubleshoot the issue
 - Software version
 - Steps required to reproduce the problem
 - Business impact of the issue

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Logging Service Requests

You may research an issue on My Oracle Support, but may be unable to locate a solution. In this case, you should log a service request (SR) through My Oracle Support. You can log an SR by clicking the Service Request tab on the My Oracle Support home page.

The first step in creating an SR is the selection of a CSI number and SR profile. After the required profile information has been submitted, My Oracle Support gathers some specifics about the problem, including the problem type, error message, brief summary of the issue, and language preference. My Oracle Support performs a search by using this information and attempts to find a solution.

The search conducted during this phase may provide different results than the searches you have performed earlier. Both searches retrieve notes and BUGs from the same database; however, the search engines and weighting are slightly different. Because the search results can differ, it is important that the search results are reviewed during the SR creation process, even if previous searches have been conducted by using the My Oracle Support search engine.

Logging Service Requests (continued)

If the search results fail to resolve the issue, the SR creation process continues with a series of questions and requests for information. After the questions are answered, the SR is submitted electronically and routed to a support representative who analyzes the issue further. Any files, screenshots, or other additional information must be uploaded immediately after the SR is logged by using the upload utility provided in the SR section of My Oracle Support.

You must ensure that the following items are clearly documented in the SR. By providing the following information, you can equip the support representative effectively to prioritize and work on the issue:

- Clear explanation of the problem, including exact error messages
- Explanation of the steps taken to troubleshoot the problem and the findings
- Exact versions of the software
- Steps required to reproduce the problem
- Business impact of this issue, including milestones, dates, and costs

Each SR is assigned a unique identifier called the *SR number*. When you log an SR, My Oracle Support provides you with the SR number (or your support representative advises you about the SR number if you log the SR by telephone). The support representative subsequently receives the SR in his or her queue through an automated allocation process that Oracle Support Services uses to distribute all phone and Web-sourced service requests. This automated process ensures that all SRs are assigned to the support representative who is best able to work on the specific issue that is being reported.

Note: For more information, refer to My Oracle Support Technical Note 166650.1 (“Working Effectively with Global Customer Support”).

Managing Patches

Kinds of patches

- Interim patches
 - For specific issues
 - No regression testing
- CPUs (Critical Patch Updates)
 - Critical security issues
 - Regression testing
 - Does not advance version number
- Patch releases



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Managing Patches

You can apply different kinds of patches at different times for different reasons.

- Interim patches (also known as *one-off* or *one-of patches*) are created to solve a specific problem. They do not go through a full regression test. Interim patches are typically installed with the `opatch` utility. The Enterprise Manager Patching Wizard can help automate the patching process by downloading, applying, and staging the patches. This wizard uses the `opatch` utility in the background.
- CPU patches (Critical Patch Update patches) include security patches and dependent non-security patches. The CPU patches are cumulative, which means fixes from previous Oracle security alerts and critical patch updates are included. It is not required to have previous security patches applied before applying the CPU patches. However, you must be on the stated patch set level. CPU patches are for a specific patch release level (such as 10.2.0.3). CPU patches are installed with the `opatch` utility or through EM Patching Wizard. The CPU patches are issued quarterly. CPU patches and interim patches can also be removed from your system with `opatch rollback -id <patch id>`. Oracle does extensive testing of Critical Patch Updates with our own applications, as well as running regression tests for the Critical Patch Updates themselves. To verify that a patch has been applied, query the inventory with `opatch -lsinventory` and see if the patch is listed.

Applying a Patch Release

- Patch releases are fully tested product fixes that:
 - Do not include new functionality
 - Affect only the software residing in your Oracle home on installation
 - Contain individual bug fixes
 - Carry version numbers
- To apply a patch:
 1. Determine your Oracle software environment.
 2. Set your My Oracle Support login credentials.
 3. Stage the patch release.

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Applying a Patch Release

Software management involves keeping your Oracle software up-to-date with the latest product fixes. Periodically, Oracle issues patch releases (product fixes) for its software. Patch releases are fully tested product fixes only; they do not include new functionality. Application of a patch release affects only the software residing in your Oracle home, with no upgrade or change to the database.

Patches are individual bug fixes. Patch sets are a collection of bug fixes up to the time of the patch set release. All patch and patch set releases carry version numbers. For example, if you bought Oracle Database 11g Release 11.1.0.2, an available patch set is 11.1.0.3. Every patch or patch set also has a patch number to identify it. Every patch release has an associated README file that describes its bug fixes. The README also has instructions for manually applying the patch.

Enterprise Manager enables you to find the latest patch release on the My Oracle Support Web site and download it to your Oracle home.

Using the Patch Advisor

The screenshot shows the Oracle Patch Advisor interface. At the top, there is a header with the title 'Patch Advisor' and a 'Logged In As' link. Below the header, there is a section titled 'Critical Security Patches' with a table. The table has columns for 'Impact', 'Abstract', 'Affected Hosts', and 'Affected Home'. A note says 'No patch advisories are currently applicable to your installation at this point in time'. Below this, there is a section titled 'Patch Recommendations by Feature' with a table. The table has columns for 'Select Patch Number', 'Created On', 'Description', 'Impacted Feature', and 'README'. There are three rows in the table. At the bottom of this section, there is a note: 'TIP It is recommended to check patch prerequisites before applying patches.' Below the table, there is a section titled 'Related Links' with a list of links: 'Patch Prerequisites', 'Database Feature Usage', 'Interim Patches Applied', 'Stage Patch', and 'Patching Setup'.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using the Patch Advisor

The Patch Advisor shows you Critical Patch Updates and recommended patches for your system. The recommendation can be based on features usage, or you can display all available patches. The Patch Setup must be configured, and the RefreshFromMetalink job must run before the patches are visible.

Click Patch Setup in the Related links section of the Patch Advisor page to navigate to the Patch Setup page.

Using the Patch Wizard

Select Patches

Select the Patches to apply. Click on "Add Patches" to search and select patches from Metalink or Software Library.

Target List

Instance Name	database
Target Type	Database Instance
Release	11.1.0.4.0
Host	stadi29.us.oracle.com
Staging Location	%oracle_home%/EMStagedPatches

This is the directory on the host where the updates will be staged.

Patches

Software Update Name	Patch Number	Created On	Type	Product	Platform	Release	Interim Patch Applicable On	Description
p4751921_11.1.0.4.0_46_94804751921		2007-02-14 00:00:00.0	Patch	Oracle Database	Linux x86	11.1.0.4.0		A useful Patch

Post Patch SQL to apply

Default (for Critical Patch Updates and Patchsets)

Custom SQL File Path Specify the file location on the host (e.g., %oracle_home%/files/patch.sql).

None

(Cancel) Step 1 of 5 (Next)

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using the Patch Wizard

When you click Apply Patch in the Database Software Patching section of the “Software and Support” page, the Patch Wizard is invoked.

The first step is Select Patches. Click Add Patch to select more patches to apply in this scheduled run.

The Target List step is reserved for patching RAC and is skipped in Oracle Database 11g.

The Library Step Properties are skipped unless the customer has customized the deployment procedures with custom variables. In the latter case, the Library Step properties are not skipped, and the user enters values for the custom variable.

In the next step, you provide the credentials for running the patch job and for determining if the job should run immediately or at a later time.

You then review the job and submit it.

Applying a Patch

Search And Select Patches

Search Metalink
 Search Software Library

Search

Patch Number	<input type="text"/>
Product Family	Oracle Database
Product	Oracle Database
Release	11.1.0.5.0
Patch Type	All Patches
Platform	Any
Language	Any

[Select All](#) | [Select None](#)

Select	Software Update Name	Patch Number	Created On	Type	Product	Platform	Release	Interim Patch Applicable On	Description	README
<input type="checkbox"/>	p6037441_11.1.0.5.0_46_9480	6037441		Patch	Oracle Database	Linux x86		11.1.0.5.0		<input type="button" value="View"/>

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Applying a Patch

You can find and apply a patch, CPU, or patch release by using the “Software and Support” page.

Staging a Patch

To look up patches at OracleMetaLink, enter search criteria and click Search. This may take a few moments depending on the number of matches found. From Search Results, select the patch to be applied and click Next. For advanced features like multiple patch application, patch flow customization, sudo and PAM support please use the "Deployment Procedures" functionality. For details on Deployment Procedures, consult the relevant [documentation](#)

Search by Number

* Patch Number [Search by Criteria](#)

Platform (If you run a 32-bit Oracle product on a 64-bit operating system, choose a 32-bit platform.)

Language

[Search](#) [Clear](#)

Search Results

	View Details	View ReadMe							
Select	Patch Number	Created On	Type	Product	Platform	Release	Interim Patch Applicable On	Description	Status
<input checked="" type="radio"/>	6037441	May 11, 2007	Patch	Oracle Database	Linux x86		11.1.0.5.0	DUMMY BUG FOR DUMMY PATCH UPLOAD TO ARU FOR BETA TESTING	available

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Staging a Patch

When you click Stage Patch in the Database Software Patching section of the “Software and Support” page, the Patch Wizard is invoked.

The first step is to select the patch either by number or by criteria.

You then select the destination. In this step, you can choose from a list of available targets.

In the third step, provide the credentials of the OS user that is to do the patching. It is recommended that this be the user that owns the software installation.

In the next step, you can choose either to stage the patch or to stage and apply the patch.

The fifth step schedules the job.

The final step enables you to review and submit the patch job.

The staged patches are stored in the `$ORACLE_HOME/EMStagedPatches_<sid>` directory on UNIX and Linux platforms, and in the `%ORACLE_HOME%\EMStagedPatches_<sid>` directory on Windows platforms.

Online Patching: Overview

For a bug fix or diagnostic patch on a running Oracle instance, online patching provides the ability to do the following:

- Install
- Enable
- Disable



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Online Patching: Overview

Online patching provides the ability to install, enable, and disable a bug fix or diagnostic patch on a live, running Oracle instance. Using online patching is the recommended solution for avoiding down time when applying online patches. Oracle provides the capability to perform online patching with any Oracle database using the `opatch` command-line utility. Online patches can be provided when the changed code is small in scope and complexity (for example, with diagnostic patches or small bug fixes).

Installing an Online Patch

- Applying an online patch does not require instance shutdown, relinking of the Oracle binary, or instance restart.
- OPatch can be used to install or uninstall an online patch.
- OPatch detects conflicts between two online patches, as well as between an online patch and a conventional patch.
- To determine if a patch is an online patch:

```
opatch query -is_online_patch <patch location>
OR
opatch query <patch location> -all
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Installing an Online Patch

Unlike traditional patching mechanisms, applying an online patch does not require instance shutdown or restart.

Similar to traditional patching, you can use OPatch to install an online patch.

You can determine whether a patch is an online patch by using the following commands:

```
opatch query -is_online_patch <patch location> or
opatch query <patch location> -all
```

Note: The patched code is shipped as a dynamic/shared library, which is then mapped to memory by each Oracle process.

Benefits of Online Patching

- No down time and no interruption of business
- Extremely fast installation and uninstallation times
- Integrated with OPatch:
 - Conflict detection
 - Listed in patch inventory
 - Works in RAC environment
- Persist across instance shutdown and startup

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Benefits of Online Patching

You do not have to shut down your database instance while you apply the online patch. Unlike conventional patching, online patching enables fast installation and uninstallation. Because online patching uses OPatch, you get all the benefits that you already have with conventional patching that uses OPatch. It does not matter how long or how many times you shut down your database—an online patch always persists across instance shutdown and startup.

Conventional Patching and Online Patching

Conventional Patches	Online Patches
Require down time to apply or remove	Do not require down time to apply or remove
Installed and uninstalled via OPatch	Installed and uninstalled via OPatch
Persist across instance startup and shutdown	Persist across instance startup and shutdown
Take several minutes to install or uninstall	Take only a few seconds to install or uninstall

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Conventional Patching and Online Patching

Conventional patching basically requires a shutdown of your database instance.

Online patching does not require any down time. Applications can keep running while you install an online patch. Similarly, online patches that have been installed can be uninstalled with no down time.

Online Patching Considerations

- Online patches are supported on the following platforms:
 - Linux x86 32/64
 - HP Itanium
 - Sun Sparc Solaris 64
 - AIX
 - Windows x86 32/64
- Some extra memory is consumed.
 - Exact amount depends on:
 - Size of patch
 - Number of concurrently running Oracle processes
 - Minimum amount of memory: Approximately one OS page per running Oracle process

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Online Patching Considerations

One operating system (OS) page is typically 4 KB on Linux x86 and 8 KB on Solaris SPARC64. With an average of approximately one thousand Oracle processes running at the same time, this represents around 4 MB of extra memory for a small online patch.

Online Patching Considerations

- There may be a small delay (a few seconds) before every Oracle process installs or uninstalls an online patch.
- Not all bug fixes and diagnostic patches are available as an online patch.
- Use online patches in situations when down time is not feasible.
- When down time is possible, you should install all relevant bug fixes as conventional patches.

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Online Patching Considerations (continued)

A vast majority of diagnostic patches are available as online patches. For bug fixes, it really depends on their nature. Not every bug fix or diagnostic patch is available as an online patch. But the long-term goal of the online-patching facility is to provide online-patching capabilities for Critical Patch Updates.

Note: You must uninstall the online patch before applying the conventional patch.

Quiz

Which of the following statements are true about online patches?

1. Can be installed using OPatch
2. Require down time to apply
3. Persist across instance startup and shutdown
4. Do not require down time to remove

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answers: 1, 3, and 4

Summary

In this lesson, you should have learned how to:

- Use the Support Workbench
- Work with Oracle Support
- Search My Oracle Support
- Log service requests
- Manage patches
 - Apply a patch release
 - Stage a patch release



Copyright © 2009, Oracle. All rights reserved.

Practice 18 Overview: Using EM Tools for Alerts and Patches

This practice covers using the Support Workbench to investigate a critical error.



Copyright © 2009, Oracle. All rights reserved.

Appendix A

Practices and Solutions

Table of Contents

Practices for Lesson 1	4
Practice 1-1: Exploring the Oracle Database Architecture	5
Practices for Lesson 2	9
Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure	10
Practice 2-2: Installing the Oracle Database 11g Software	18
Practices for Lesson 3	21
Practice 3-1: Creating an Oracle Database	22
Practices for Lesson 4	30
Practice 4-1: Managing the Oracle Instance	31
Practice 4-2: Testing Oracle Restart and Your Database.....	37
Practices for Lesson 5	39
Practice 5-1: ASM Fast Mirror Resync	40
Practice 5-2: Using ASMCMD	45
Practices for Lesson 6	48
Practice 6-1: Configuring the Oracle Network to Access Another Database	49
Practice 6-2: Creating an Alternate Listener.....	53
Practices for Lesson 7	56
Practice 7-1: Viewing Database Storage Structure Information	57
Practice 7-2: Creating a Tablespace.....	61
Practices for Lesson 8	66
Practice 8-1: Creating and Using a Profile	67
Practice 8-2: Creating Roles	69
Practice 8-3: Creating and Configuring Users.....	71
Practices for Lesson 9	75
Practice 9-1: Resolving Lock Conflicts	76
Practices for Lesson 10	80
Practice 10-1: Managing Undo Data.....	81
Practices for Lesson 11	86
Practice 11-1: Configuring Database Auditing.....	87
Practices for Lesson 12	90
Practice 12-1: Database Maintenance.....	91
Practices for Lesson 13	100
Practice 13-1: Managing Performance.....	101
Practices for Lesson 14	110
Practice 14-1: Configuring Your Database for Recovery.....	111
Practices for Lesson 15	116
Practice 15-1: Performing Database Backups.....	117
Practices for Lesson 16	123
Practice 16-1: Preparing Practice Environment.....	124
Practice 16-2: Recovering from the Loss of a Data File.....	125
Practice 16-3: Recovering from the Loss of a File in the SYSTEM Tablespace	128
Practice 16-4: Recovering from the Loss of a Control File	131
Practices for Lesson 17	135
Practice 17-1: Moving Data Using Data Pump.....	136

Practice 17-2: Loading Data Using SQL*Loader.....	142
Practices for Lesson 18	146
Practice 18-1: Investigating a Critical Error	147

Practices for Lesson 1

Background: In this practice, you review concepts about Oracle architecture components and answer questions to test your knowledge of the concepts learned in the lesson.

Practice 1-1: Exploring the Oracle Database Architecture

Fill in the blanks with the correct answers.

1) The two main components of a basic Oracle Database system:

_____ and _____

Hint: see page 1-6

2) The Instance consists of _____ and
_____ processes.

Hint: see page 1-6

3) A session is a connection between the _____ process and the
_____ process.

Hint: see page 1-8

4) Name the main components of the SGA:

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____

Hint: see page 1-9

Practice 1-1: Exploring the Oracle Database Architecture (continued)

5) List six of the many background processes an Oracle Database instance might have:

- _____
- _____
- _____
- _____
- _____
- _____

Hint: see page 1-21

6) The _____ process writes the dirty buffers to the data files.

Hint: see page 1-23

7) The _____ process writes the redo entries to the online redo log files.

Hint: see page 1-25

8) The primary files associated with an Oracle database are:

- _____
- _____
- _____

Additional important files are:

- _____
- _____
- _____
- _____
- _____

Hint: see page 1-33

Practice 1-1: Exploring the Oracle Database Architecture (continued)

9) The logical storage structures of an Oracle database are:

- _____
- _____
- _____
- _____
- _____

Hint: see page 1-35

10) The _____ process copies the redo log files to an archive destination.

Hint: see page 1-31

11) The _____ contains data and control information for a server or background process.

Hint: see page 1-17

12) The logical tablespace structure is associated with the physical _____ files on disk.

Hint: see page 1-35

13) LGWR writes when:

- _____
- _____
- _____
- _____

Hint: see page 1-25

Practice 1-1: Exploring the Oracle Database Architecture (continued)

14) State whether the following statements are true or false.

- a) The SGA includes the Database buffer cache and Redo log buffer. _____
- b) Each server process has its own PGA. _____
- c) The buffers in the database buffer cache are organized in two lists: the most recently used list and the least recently used (LRU) list. _____
- d) User processes run the application or tool that connects to an Oracle Instance.

- e) Oracle Database processes include server processes and background processes.

- f) Checkpoints are recorded in log file headers. _____

Hint: see pages 1-9, 1-10, 1-13, 1-20, 1-21, 1-27

Practices for Lesson 2

Background: In the practices of this course, you assume the role of a database administrator (DBA). The operating system (OS) accounts on your computer are:

- The `oracle` user with a password of `oracle`
- The `root` user with a password of `oracle`

The system administrator has set up the OS so that it is ready for your Oracle software installation. You are performing two installations. The first installation is the Oracle Grid Infrastructure for a standalone server. The second installation is the Oracle Database 11g software.

The installation media is staged at:

- `/stage/11.2.0/clusterware/Disk1`. for Oracle Grid Infrastructure
- `/stage/11.2.0/database/Disk1`. for Oracle Database 11g

Perform the following tasks as the default `oracle` OS user, unless otherwise indicated.

Note: Completing this practice is critical for all following practice sessions.

Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure

In this practice you install the Oracle Grid Infrastructure for a standalone sever. During the installation, you configure the +DATA ASM disk group that will be used for storing database files for your database. After the installation is complete, you configure the +FRA ASM disk group that will be used for database backups and other database files for your database.

- 1) Start the Oracle Universal Installer (OUI) for the Oracle Grid Infrastructure. As the `oracle` user, navigate to the `/stage/11.2.0/clusterware/Disk1` directory and enter `./runInstaller`.

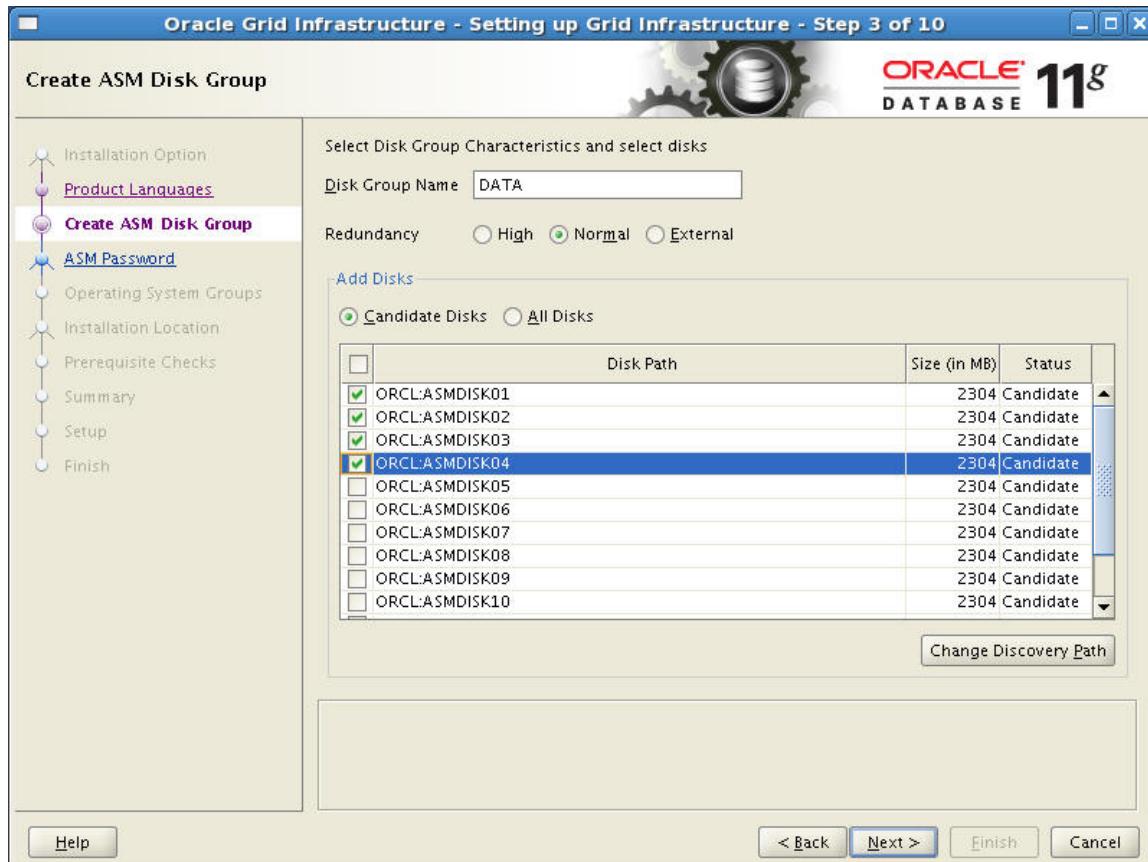
- a) Right-click the desktop and click Open Terminal to open a terminal window. Then enter the following:

```
$ cd /stage/11.2.0/clusterware/Disk1
$ ./runInstaller
```

- 2) On the Select Installation Option page, select the Install and Configure Grid Infrastructure for a standalone server option and click Next.
- 3) On the Product Languages page, select all the available languages and click Next.

Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure (continued)

- 4) On the Create ASM Disk Group page, ensure the Disk Group Name is DATA and Redundancy is Normal. Select the first four disk groups (ORCL:ASMDISK01, ORCL:ASMDISK02, ORCL:ASMDISK03, and ORCL:ASMDISK04) and then click Next.



- 5) On the Specify ASM Password page, select the option to use the same passwords for both SYS and SYSTEM accounts. Enter **oracle_4U** as the password and click Next.
- 6) The Privileged Operating System Groups page is next. Because your installation is for a standalone server, the same operating system group (dba) can be used for all of the administration groups shown.

Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure (continued)

a) Select **dba** for all three options and click **Next** to continue.



b) A warning appears because we have specified the same operating group for all administrator groups. This is expected, so click **Yes** to continue.

Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure (continued)

7) On the Installation Location page, ensure that the value for Oracle Base is **/u01/app/oracle**. For Software Location, enter **/u01/app/oracle/product/11.2.0/grid**. Click **Next**.



8) On the Create Inventory page, accept all the defaults and click **Next** to continue.

9) The Perform Prerequisite Checks page is next. The OUI checks to make sure that your environment meets the minimum requirements for this installation. In many cases if a prerequisite check fails, the OUI can generate a fixup script to fix the problem. In our classroom, all prerequisites have been met so no issues are found and the OUI automatically advances to the next page.

10) On the Summary page, review the settings and information, and then click **Finish**.

11) The Setup page appears showing the progress of the installation and the status of the individual tasks being performed. When the Execute Configuration scripts window appears, follow the steps listed in the window.

- Open a terminal window and log in as `root`.

```
$ su -  
Password:  
#
```

- Run the scripts shown in the Execute Configuration scripts window.

Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure (continued)

i) The first script is `/u01/app/oraInventory/orainstRoot.sh`.

```
# /u01/app/oraInventory/orainstRoot.sh
Changing permissions of /u01/app/oraInventory.
Adding read,write permissions for group.
Removing read,write,execute permissions for world.

Changing groupname of /u01/app/oraInventory to oinstall.
The execution of the script is complete.
#
```

ii) The second script is

`/u01/app/oracle/product/11.2.0/grid/root.sh`. Accept the default of `/usr/local/bin` for the local bin directory by pressing Enter when prompted.

```
# /u01/app/oracle/product/11.2.0/grid/root.sh
Running Oracle 11g root.sh script...

The following environment variables are set as:
  ORACLE_OWNER= oracle
  ORACLE_HOME=  /u01/app/oracle/product/11.2.0/grid

Enter the full pathname of the local bin directory:
[/usr/local/bin]:
  Copying dbhome to /usr/local/bin ...
  Copying oraenv to /usr/local/bin ...
  Copying coraenv to /usr/local/bin ...

Creating /etc/oratab file...
Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root.sh script.
Now product-specific root actions will be performed.
2009-07-08 09:35:07: Checking for super user privileges
2009-07-08 09:35:07: User has super user privileges
2009-07-08 09:35:07: Parsing the host name
Using configuration parameter file:
/u01/app/oracle/product/11.2.0/grid/crs/install/crsconfig_params
Creating trace directory
LOCAL ADD MODE
Creating OCR keys for user 'oracle', privgrp 'oinstall'..
Operation successful.
CRS-4664: Node edrsr12p1 successfully pinned.
Adding daemon to inittab
CRS-4123: Oracle High Availability Services has been started.
ohasd is starting
```

Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure (continued)

```
edrsr12p1      2009/07/08 09:35:39
/u01/app/oracle/product/11.2.0/grid/cdata/edrsr12p1/backup_200
90708_093539.olr
Successfully configured Oracle Grid Infrastructure for a
Standalone Server
#
```

- c) Click **OK** in the Execute Configuration scripts window. The OUI continues with the remaining installation tasks.

12) Click **Close** on the Finish page to complete the installation of the Oracle Grid Infrastructure for a standalone server.

13) The next step is to configure the +FRA disk group. In a terminal window, logged in as **oracle**, perform the following steps:

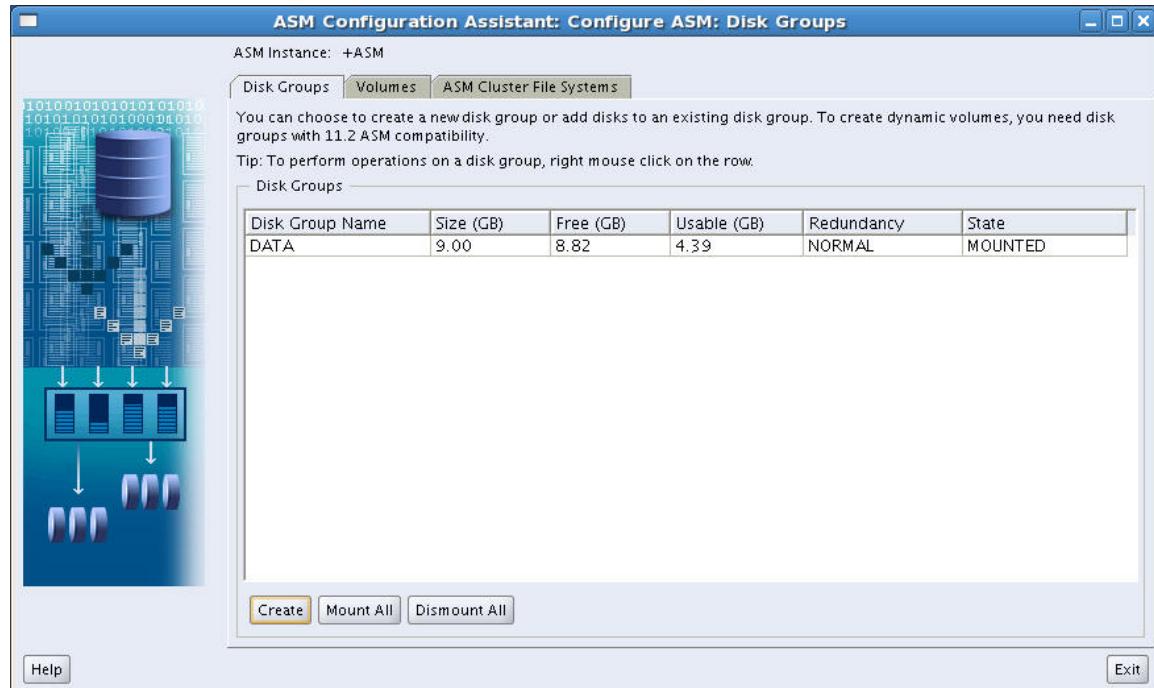
- a) Use the **oraenv** utility to set the environment for the terminal session. Enter **+ASM** when prompted for the **ORACLE_SID**:

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
```

- b) Start the ASM Configuration Assistant by entering **asmca** at the command line.

```
$ asmca
```

- c) The ASM Configuration Assistant opens displaying the current disk groups for the +ASM instance. Click **Create**.

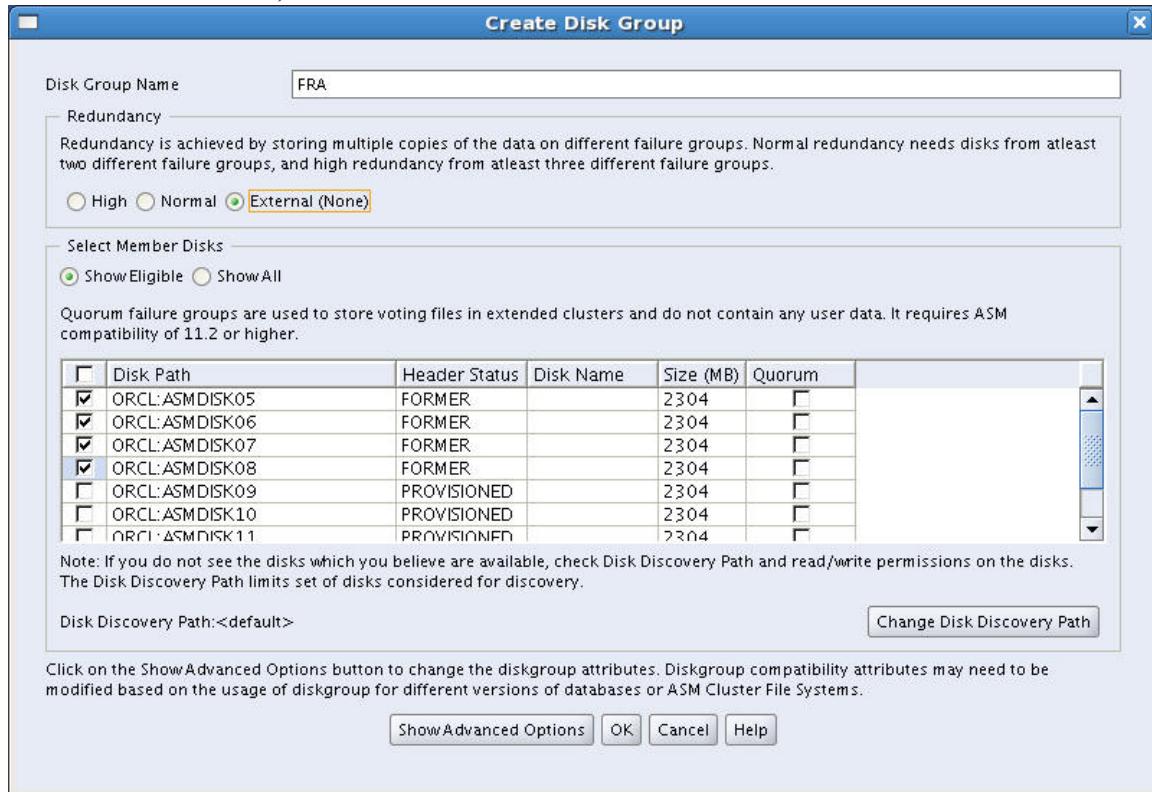


Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure (continued)

d) Enter **FRA** for the Disk Group Name. Select External (None) for redundancy.

Select disk groups **ORCL:ASMDISK05**, **ORCL:ASMDISK06**,

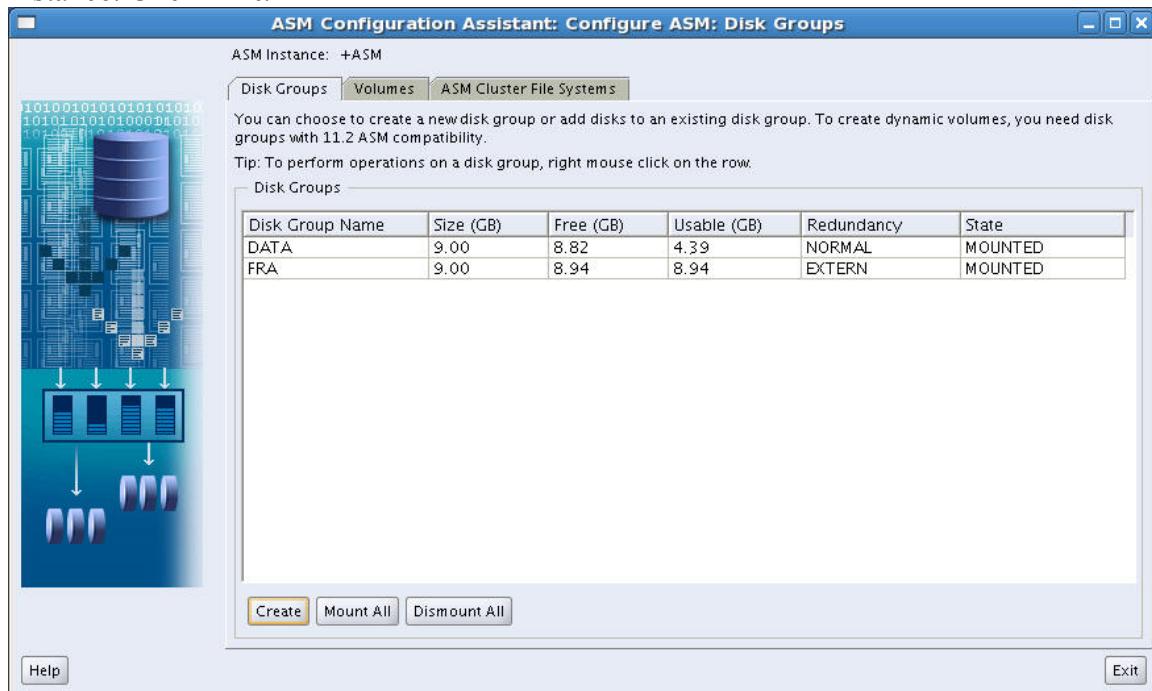
ORCL:ASMDISK07, and **ORCL:ASMDISK08** and click **OK**.



e) Click **OK** in the DiskGroup: Creation window when it appears.

Practice 2-1: Installing and Configuring the Oracle Grid Infrastructure (continued)

f) Notice that now there are two disk groups (DATA and FRA) listed for the +ASM instance. Click **Exit**.



g) Click **Yes** when asked if you really want to quit this application.
14) Close any open terminal windows.

Practice 2-2: Installing the Oracle Database 11g Software

The next step is to install the Oracle Database 11g software.

- 1) Open a new terminal window and install the Oracle database software as the `oracle` user. Navigate to the `/stage/11.2.0/database/Disk1` directory, and start the Oracle Universal Installer (OUI) by entering `./runInstaller`.
 - a) Right-click the desktop and click Open Terminal to open a terminal window. Then enter the following:

```
$ cd /stage/11.2.0/database/Disk1
$ ./runInstaller
```

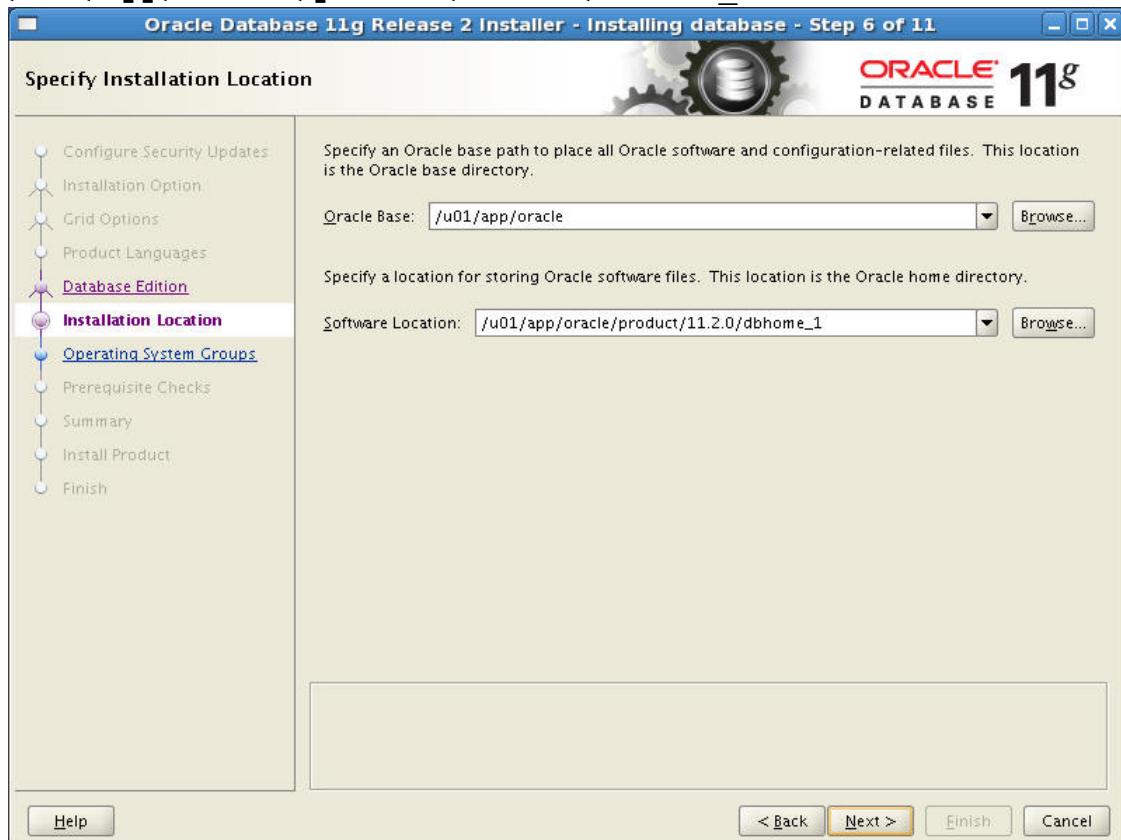
- 2) The Configure Security Updates page is the first to appear. In your real-world environment, you would enter your email address and My Oracle Support password; however, because the classroom is an isolated environment, please leave the email and password fields blank. Deselect the option to receive security updates from My Oracle Support and click **Next**.
- 3) Click **Yes** in the Email Address Not Specified warning message that appears.
- 4) On the Installation Option page, select the **Install database software only** option and click **Next**.



- 5) Ensure that **Single instance database installation** is selected on the Install Type page and click **Next**.
- 6) On the Product Languages page, select all the available languages and click **Next**.

Practice 2-2: Installing the Oracle Database 11g Software (continued)

- 7) On the Select Database Edition page, ensure **Enterprise Edition (3.95GB)** is selected and click **Next**.
- 8) On the Installation Location page, ensure that the value for Oracle Base is `/u01/app/oracle`. For Software Location, enter `/u01/app/oracle/product/11.2.0/dbhome_1`. Click **Next**.



- 9) Select **dba** for both the OSDBA and OSOPER groups on the Privileged Operating System Groups page and click **Next**.
- 10) The OUI then performs prerequisite checks. No problems should be found and the Summary page should appear next. Click **Finish**.
- 11) The Install Product page appears, showing you the progress of the installation and the status for each individual task being performed. When the Execute Configuration scripts window appears, follow the steps listed in the window.
 - a) Open a terminal window and log in as `root`.

```
$ su -  
Password:  
#
```

- b) Run the script shown in the Execute Configuration scripts window. Accept the default for the local `bin` directory and do not overwrite any files (you can just press [Enter] because the default option is to not overwrite).

Practice 2-2: Installing the Oracle Database 11g Software (continued)

```
# /u01/app/oracle/product/11.2.0/dbhome_1/root.sh
Running Oracle 11g root.sh script...

The following environment variables are set as:
ORACLE_OWNER= oracle
ORACLE_HOME=  /u01/app/oracle/product/11.2.0/dbhome_1

Enter the full pathname of the local bin directory:
[/usr/local/bin]:
The file "dbhome" already exists in /usr/local/bin.  Overwrite
it? (y/n)
[n]:
The file "oraenv" already exists in /usr/local/bin.  Overwrite
it? (y/n)
[n]:
The file "coraenv" already exists in /usr/local/bin.
Overwrite it? (y/n)
[n]:

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root.sh script.
Now product-specific root actions will be performed.
Finished product-specific root actions.
#
```

- c) Click **OK** on the Execute Configuration scripts window.
- 12) Click **Close** on the Finish page to complete the installation of the Oracle Database 11g software.

Practices for Lesson 3

Background: You are about to begin creating your first Oracle database. You anticipate that several similar databases will be needed in the near future. Therefore, you decide to create your `orcl` database, as well as a database template and the database creation scripts. Locate the scripts in the `/home/oracle/labs` directory (which is the directory that you use most often throughout this course).

Practice 3-1: Creating an Oracle Database

In this practice, you create the `orcl` database. You use the Database Configuration Assistant (DBCA) to create the database.

- 1) Start the Database Configuration Assistant (DBCA).
 - a) Open a terminal window as the `oracle` user by right-clicking your desktop and selecting **Open Terminal**.
 - b) Set your `ORACLE_HOME` environment variable by using `oraenv`. Enter `orcl` for the SID and then enter
`/u01/app/oracle/product/11.2.0/dbhome_1` for `ORACLE_HOME`.
Note: You enter the full `ORACLE_HOME` path at this time because the `orcl` database does not yet exist. After the database is created, you will only have to enter `orcl` as the SID and it will determine the correct `ORACLE_HOME`.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/11.2.0/dbhome_1
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
```

- c) To start the DBCA, enter:

```
$ dbca
```

- 2) Click **Next** on the Welcome page to begin the `orcl` database creation.
- 3) On the Operations page, select **Create a Database**, and then click **Next**.
- 4) On the Database Templates page, select the **General Purpose or Transaction Processing** template.
 - a) Click **Show Details** and answer the following questions:
 - i) **Question 1:** How many control files are created?

Answer: Two

Note: The location will change later in this practice when we choose to use ASM as our storage technique.

- ii) **Question 2:** How many redo log groups are created?

Answer: Three

Note: The location will change later in this practice when we choose to use ASM as our storage technique.

- iii) **Question 3:** What is the database block size (`db_block_size`)?

Answer: 8 KB

Practice 3-1: Creating an Oracle Database (continued)

iv) **Question 4:** What is the value of Sample Schemas?

Answer: Sample Schemas is set to False.

Note: You will change this setting later in this practice so that the HR sample schema is included in your database.

v) **Question 5:** What is the template default for the Database Character Set?

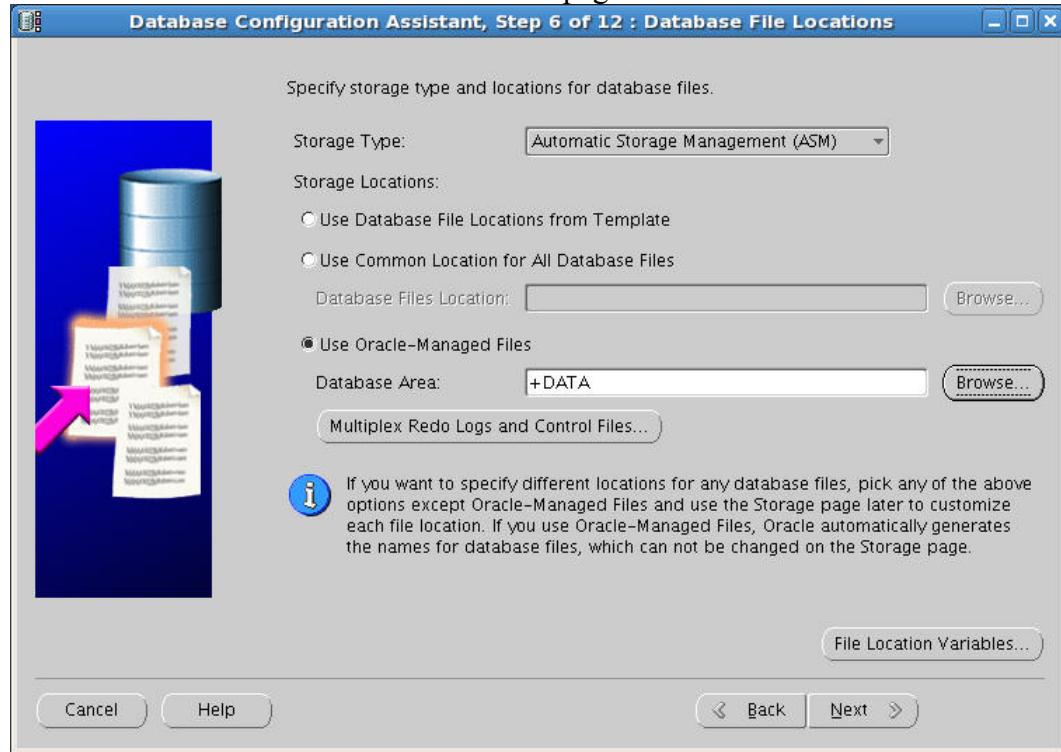
Answer: WE8MSWIN1252

Note: You will change this setting later in this practice to use a Unicode database character set.

- b) Click **Close** to close the Template Details window.
- c) Click **Next** on the Database Templates page to continue the database creation process.
- 5) On the Database Identification page, enter **orcl.example.com** as Global Database Name. The SID defaults to the database name **orcl**. Click **Next**.
- 6) On the Management Options page, ensure that the following items are selected:
 - a) On the Enterprise Manager tab, ensure that both **Configure Enterprise Manager** and **Configure Database Control for local Management** are selected.
 - b) On the Automatic Maintenance Tasks tab, ensure that **Enable automatic maintenance tasks** is selected.
 - c) Click **Next** to continue.
- 7) On the Database Credentials page, select **Use the Same Password for All Accounts** and enter **oracle_4U** as Password and Confirm Password. Then click **Next**.
- 8) On the Storage Options page, specify ASM as the storage type and choose the **+DATA** disk group as the storage location.
 - a) Select **Automatic Storage Management (ASM)** for Storage Type
 - b) The storage location defaults to Use Oracle-managed Files. Click the **Browse** button for Database Area.
 - c) In the Select Disk Group window, ensure that the **DATA** disk group is selected and click **OK**.

Practice 3-1: Creating an Oracle Database (continued)

d) Click **Next** on the Database File Locations page.



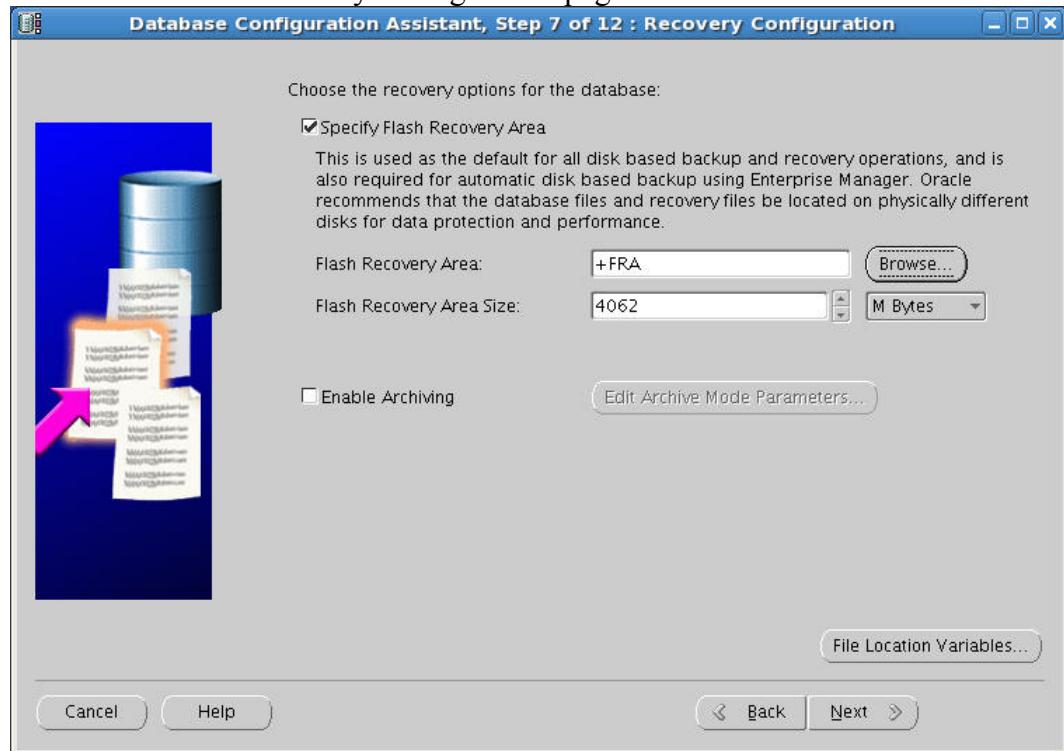
e) The ASM Credentials window appears. Enter the password you specified during ASM installation and configuration (for your classroom environment this should be **oracle_4U**) and click **OK**.

9) On the Recovery Configuration page, configure the Fast Recovery Area.
Note: The Flash Recovery Area is now the Fast Recovery Area but in this release the pages in Enterprise Manager still refer to it as Flash Recovery Area. Watch for this to change in future releases as the product goes through this change.

- Select **Specify Flash Recovery Area**
- Click the **Browse** button for Flash Recovery Area.
- In the Select Disk Group window, select the **FRA** disk group and click **OK**.

Practice 3-1: Creating an Oracle Database (continued)

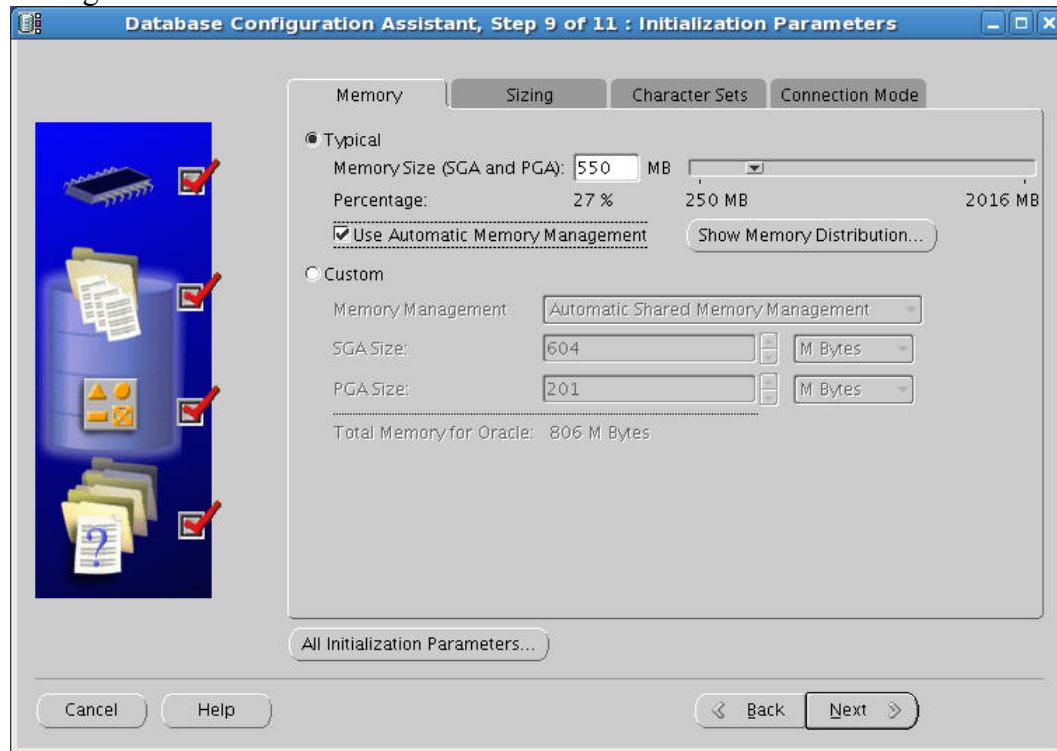
d) Click **Next** on the Recovery Configuration page.



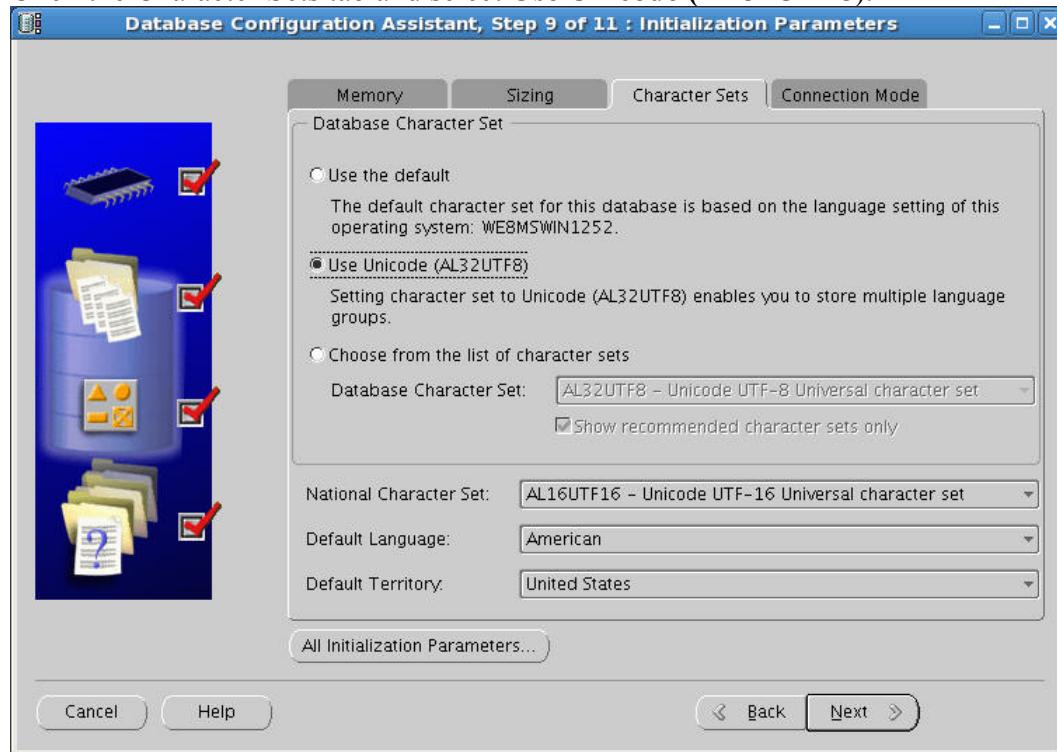
- 10) On the Database Content page, select **Sample Schemas**, and then click **Next**.
- 11) On the Memory tabbed page of the Initialization Parameters page, select **Typical** and specify a size of **550 MB** for Memory Size. Modify the character set to use **Unicode AL32UTF8**.

Practice 3-1: Creating an Oracle Database (continued)

a) Select **Typical** and set the value for **Memory Size (SGA and PGA)** to **550 MB**. Ensure **Automatic Memory Management** is selected for the Memory Management field.



b) Click the Character Sets tab and select **Use Unicode (AL32UTF8)**.



Practice 3-1: Creating an Oracle Database (continued)

- c) Review the Sizing and Connection Mode tabbed pages, but do not change any values. Then click **Next**.
- 12) On the Database Storage page, review your file names and locations. Then click **Next**.
- 13) On the Creation Options page make the following selections:
 - a) Select **Create Database**.
 - b) Select **Save as a Database Template** option. Enter **orcl** as the Name for the database template and **orcl Database Template** as the Description.
 - c) Select **Generate Database Creation Scripts** and enter **/home/oracle/labs** as the Destination Directory.
 - d) Then click **Finish**.
- 14) The Confirmation page appears.

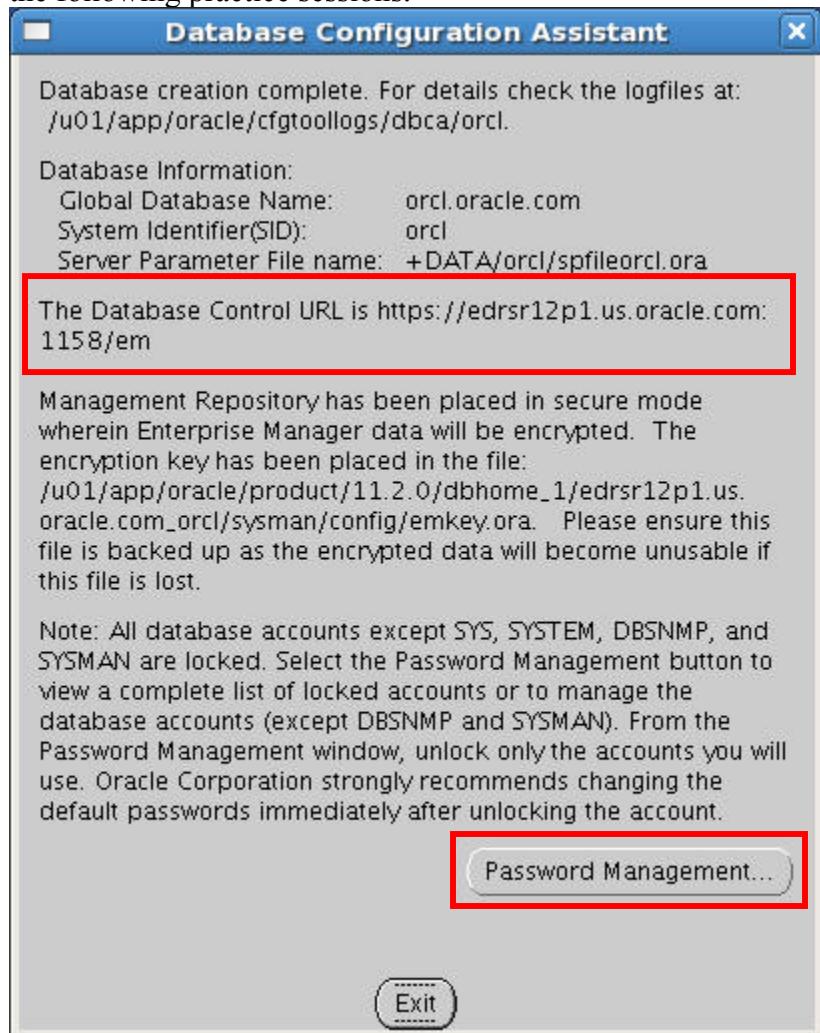
- a) Review options and parameters, specifically the ones in the table below, and click **OK**.

Name	Value
Sample Schemas	true
db_block_size	8KB
db_create_file_dest	+DATA
db_recovery_file_dest	+FRA
memory_target	550MB
Database Character Set	AL32UTF8

- b) Click **OK** to acknowledge that the template has been created.
- c) Click **OK** to acknowledge the generation of the database scripts.

Practice 3-1: Creating an Oracle Database (continued)

15) The DBCA displays the progress of the various installation steps. When the database itself has been created, the DBCA displays essential information about the database. Make note of this information. The Database Control URL will be used in several of the following practice sessions.



a) **Important:** Make note of your Database Control URL here:

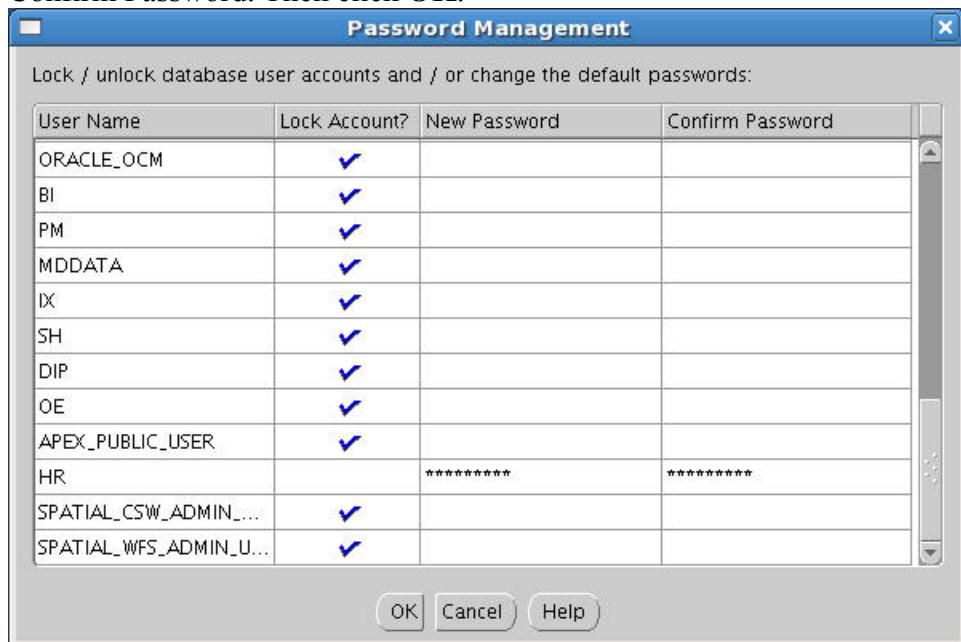
https://_____ : _____/em

You will be using this URL many times throughout the remainder of the course.

b) Click the **Password Management** button.
c) Scroll down the Password Management page until you see the **HR** username.

Practice 3-1: Creating an Oracle Database (continued)

d) Deselect **Lock Account?** and enter **oracle_4U** as the New Password and Confirm Password. Then click **OK**.



e) Click **Exit** to close the DBCA.

You have completed your task to create a database, a database template, and database generation scripts.

Practices for Lesson 4

Background: You have just installed the Oracle software and created a database. You want to ensure that you can start and stop the database and see the application data.

Practice 4-1: Managing the Oracle Instance

In this practice, you get to know the Oracle Enterprise Manager interface a little better. Using Oracle Enterprise Manager you perform the following tasks:

- View and change instance parameters
- Shut down the database
- Start up the database

You also view various instance parameters using SQL*Plus and look the text version of the alert log from a terminal window.

- 1) Invoke Enterprise Manager, and log in as the **SYS** user. Which port number does this database use? You noted this in Practice 3.

Answer: 1158

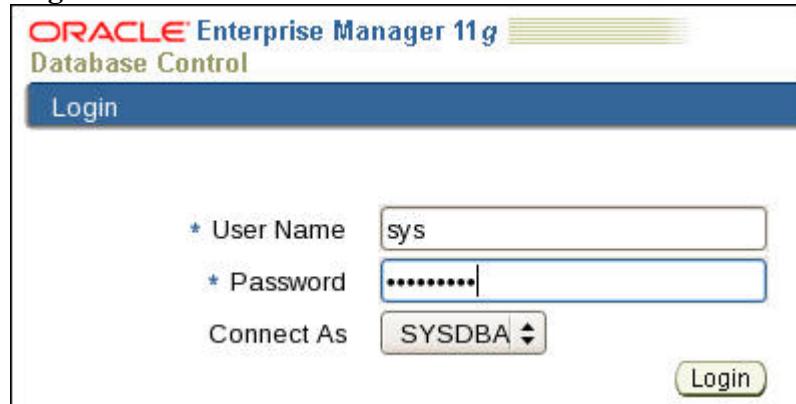
- a) Double-click the **Web Browser** icon on your desktop to open your Web browser as the **oracle** user.
- b) Enter the URL that you wrote down in Practice 3. It has the following format:

`https://hostname:portnumber/em`

Note: The first time you connect you will get a Secure Connection Failed message (or something similar) and an Alert window may appear. To get past this, you add an exception and accept the certificate.

- i) Click **OK** on the Alert window if it appears.
- ii) Click the **Or you can add an exception...** link at the bottom of the page.
- iii) A warning regarding adding exceptions appears. Click the **Add Exception...** button.
- iv) On the Add Security Exception window, click the **Get Certificate** button.
- v) The Certificate Status is displayed. Ensure that the option to permanently store this exception is selected and click the **Confirm Security Exception** button.

- c) In the Oracle Enterprise Manager login screen, enter **sys** as the User Name, enter **oracle_4U** as the Password, and select **SYSDBA** for Connect As. Then click **Login**.



The screenshot shows the Oracle Enterprise Manager 11g Database Control login interface. The title bar reads "ORACLE Enterprise Manager 11g Database Control". The main area is titled "Login". It contains three input fields: "User Name" with "sys" entered, "Password" with "*****" entered, and "Connect As" with "SYSDBA" selected. A "Login" button is located at the bottom right.

- 2) View the initialization parameters and set the **JOB_QUEUE_PROCESSES** parameter to **15**. What SQL statement is run to do this?

Practice 4-1: Managing the Oracle Instance (continued)

a) Select Server > Initialization Parameters (in the Database Configuration section).

Database Instance: orcl.oracle.com

Home	Performance	Availability	Server	Schema	Data Movement
Storage			Database Configuration		
Control Files			Memory Advisors		
Tablespaces			Automatic Undo Management		
Temporary Tablespace Groups			Initialization Parameters		
Datafiles			View Database Feature Usage		
Rollback Segments					
Redo Log Groups					

b) Enter **job** in the Name field, and then click **Go**.

Initialization Parameters

Current	SPFile			
The parameter values listed here are currently used by the running instance(s). You can change static parameters in SPFile mode.				
Name	Basic	Modified	Dynamic	Category
<input type="text" value="job"/>	<input type="button" value="All"/>	<input type="button" value="All"/>	<input type="button" value="All"/>	<input type="button" value="All"/>
Filter on a name or partial name				

c) When the **JOB_QUEUE_PROCESSES** initialization parameter appears, change its value to **15**.

d) Click **Show SQL** and note the SQL statement that is going to be run next.

Database Instance: orcl.oracle.com > Initialization Parameters > Logged in As SYS

Show SQL [Return](#)

```
ALTER SYSTEM SET job_queue_processes = 15 SCOPE=MEMORY
```

e) Click **Return**, and then click **Apply**.

3) *Question:* What is the significance of a check in the Dynamic column?
Answer: A “dynamic” parameter can be modified while the database is running.

4) Shut down the database instance by using Enterprise Manager.

- In the Enterprise Manager browser session, click the **Database** tab.
- Click the **Shutdown** button.
- For Host Credentials, enter **oracle** as Username and **oracle** as Password.
- Click **OK**. The Startup/Shutdown: Confirmation page appears.
- Click **Advanced Options** to see the mode for shutting down, but do not change the mode; it should remain as “Immediate.”
- Click **Cancel** to return to the previous page.
- Click **Yes** to confirm the shutdown operation.
- Click **Refresh**. If you see an error during the refresh, click **OK** and continue to refresh. The error will resolve itself.

Practice 4-1: Managing the Oracle Instance (continued)

i) Note that the Status of the instance is now “Down.”



Database Instance

Status **Down** Details **There has been a user-initiated shutdown.**

Host **edrsr12p1.us.oracle.com** Port **1521** SID **orcl**

Oracle Home **/u01/app/oracle/product/11.2.0/dbhome_1**

Startup **Perform Recovery**

5) Using SQL*Plus, verify that you are *not* able to connect as the HR user to a database that has been shut down.

- In the Linux command window set your environment to the **orcl** database using **oraenv**.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$
```

- Enter the following to attempt to log in to the database:

```
$ sqlplus hr
```

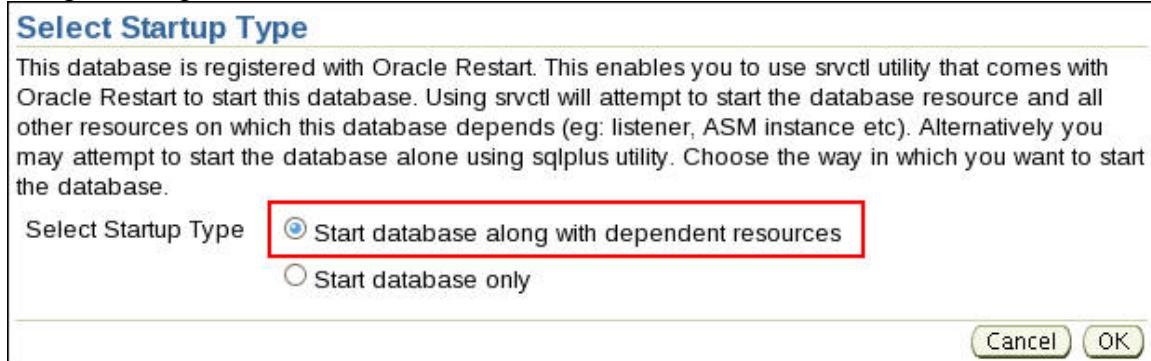
- Enter **oracle_4U** for the password.
- Note the “ORACLE not available” error message.
- Press **[Ctrl], [D]** to exit the username prompt.

6) Use Enterprise Manager to restart the database instance, and then log in as the **SYS** user again.

- In Enterprise Manager, click the **Startup** button.
- Enter **oracle** for both Username and Password in the Host Credentials region.
- Click **OK**.

Practice 4-1: Managing the Oracle Instance (continued)

d) The Select Startup Type page appears. Ensure the option to start the database along with dependent resources is selected and click **OK**.



e) The Startup/Shutdown: Confirmation page appears.

f) Click **Advanced Options** to see the modes and options available for starting up, but do not change the mode; the startup mode should remain as “Open”.

g) Click **Cancel** to return to the previous page.

h) Click **Yes** to confirm the startup operation.

i) The Startup page appears as the database is starting up. Wait for the login page to appear, at which time you can log in as **SYS** user with the **oracle_4U** password and the **SYSDBA** privilege.

Note: When you first connect you may see a Failed status for the Agent Connection to Instance. Wait a few minutes and this should go away and the database home page should appear.

7) In the alert log, view the phases that the database went through during startup. What are they?

- Select **Database > Related Links > Alert Log Contents**. Click **Go**.
- Scroll through the log and review the phases of the database during startup. Your alert log may look different from this screenshot, based on different system activities.

Jul 10, 2009 5:44:48 AM GMT+07:00	NOTIFICATION	16	admin_ddl opixe:2994:4222364190	ALTER DATABASE OPEN
Jul 10, 2009 5:44:48 AM GMT+07:00	NOTIFICATION	16	admin_ddl opixe:3065:2802784106	Completed: ALTER DATABASE MOUNT

- Note that the modes that the database goes through during startup are **MOUNT** and **OPEN**.
- Locate and view the text version of the alert log.
Connect to the database as the **system** user (password is **oracle_4U**) using SQL*Plus and query the **V\$DIAG_INFO** view. To view the text-only alert log without the XML tags, complete these steps:
 - In the **V\$DIAG_INFO** query results, note the path that corresponds to the **Diag Trace** entry.

```
SQL> select * from V$DIAG_INFO;
```

Practice 4-1: Managing the Oracle Instance (continued)

```
INST_ID NAME
-----
VALUE
-----
...
1 Diag Trace
/u01/app/oracle/diag/rdbms/orcl/orcl/trace
...
```

ii) Exit from SQL*Plus and change directory to that path.

```
$ cd /u01/app/oracle/diag/rdbms/orcl/orcl/trace
```

iii) Open the **alert_orcl.log** file with a text editor.

Note: The file will be named **alert_<sid>.log** in other databases, where **<sid>** is the instance name.

e) Try to locate the entries for the shutdown and startup performed earlier in the practice.

8) Connect to the database using SQL*Plus as **sysdba**.

Note: Remember to use **oraenv** to set your environment to the **orcl** database if you have not already done so in your terminal window.

```
$ sqlplus / as sysdba
```

9) Use the **SHOW PARAMETER** command to verify the settings for **SGA_MAX_SIZE**, **DB_CACHE_SIZE**, and **SHARED_POOL_SIZE**.

```
SQL> show parameter sga_max_size
NAME                           TYPE        VALUE
-----
sga_max_size                   big integer 552M
SQL> show parameter db_cache_size
NAME                           TYPE        VALUE
-----
db_cache_size                  big integer 0
SQL> show parameter shared_pool_size
NAME                           TYPE        VALUE
-----
shared_pool_size                big integer 0
SQL>
```

10) Check the value of **JOB_QUEUE_PROCESSES**.

Practice 4-1: Managing the Oracle Instance (continued)

```
SQL> show parameter job_queue_processes
```

NAME	TYPE	VALUE
job_queue_processes	integer	1000

```
SQL>
```

Note: Earlier in this practice, you changed the JOB_QUEUE_PROCESSES parameter to 15, but the scope of that change was for the running instance only. That change did not get saved in the SPFILE. So the next time you performed a shutdown and startup, the value reverted to the value in the SPFILE.

Practice 4-2: Testing Oracle Restart and Your Database

In this practice, you test the Oracle Restart functionality by causing your database to crash and watching for Oracle Restart to restart your database.

- 1) Use Enterprise Manager to determine whether your `orcl` database instance is currently managed by Oracle Restart.
 - a) Go to the Home page by clicking the **Database** tab.
 - b) On the Home page, look at the **High Availability** section and see that Oracle Restart is **Enabled** for your database instance.
- 2) To determine the effect of Oracle Restart, kill the LGWR process of your `orcl` database instance. What do you observe?
 - a) Set your environment variables for your `orcl` database instance.

```
$ . oraenv
ORACLE_SID = [orcl] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
Enter
```

- b) Enter `ps -ef | grep ora_lgwr_orcl` to find the process ID for the LGWR process for your database.

```
$ ps -ef | grep ora_lgwr_orcl
oracle  10478      1  0 10:54 ?          00:00:00 ora_lgwr_orcl
oracle  11371  24865  0 11:00 pts/3      00:00:00 grep
ora_lgwr_orcl
```

- c) Kill the LGWR using the `kill -9` command and the process ID you determined in the previous step. This will cause the instance to shut down.

```
$ kill -9 10478
```

- d) Enter `ps -ef | grep ora_lgwr_orcl` again to see if the LGWR process is restarted. Repeat this until you see that the LGWR has started again. Notice that the `ora_lgwr_orcl` process has a different process ID now than the process ID you used when issuing the `kill -9` command.

```
$ ps -ef | grep ora_lgwr_orcl
oracle  11849 11687  0 11:06 pts/3      00:00:00 grep
ora_lgwr_orcl
$ ps -ef | grep ora_lgwr_orcl
oracle  11855 11687  0 11:06 pts/3      00:00:00 grep
ora_lgwr_orcl
$ ps -ef | grep ora_lgwr_orcl
oracle  11946      1  0 11:06 ?          00:00:00 ora_lgwr_orcl
oracle  12034 11687  0 11:07 pts/3      00:00:00 grep
ora_lgwr_orcl
```

Practice 4-2: Testing Oracle Restart and Your Database (continued)

3) Connect to the database using SQL*Plus as sysdba to confirm that your database has restarted successfully. Query v\$instance to see the status of your database. **Note:** Remember to use oraenv to set your environment to the orcl database if you have not already done so in your terminal window.

```
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Tue Aug 18 11:16:40
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.2.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options

SQL> select status from v$instance;

STATUS
-----
OPEN

SQL>
```

Practices for Lesson 5

Background: In these practices, you explore the benefits of the ASM Fast Mirror Resync feature. You also explore the functionality of the ASM command-line utility, ASMCMD.

Practice 5-1: ASM Fast Mirror Resync

In this practice, you compare the time it takes to add an offlined disk following a non-data-loss issue. You compare the same operation once without using ASM Fast Mirror Resync, and once using it.

- 1) Determine the compatibility values for your existing ASM disk groups. What do you observe?
 - a) Set the proper environment variables for the +ASM instance. Run the query in the following screenshot to check COMPATIBLE values for both ASM and the database. You should see that the database compatibility value is set to 10.1.

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ sqlplus / as sysasm

SQL> select name,compatibility,database_compatibility from
v$asm_diskgroup;

NAME          COMPATIBILITY      DATABASE_COMPATIBILITY
-----          -----          -----
DATA           11.2.0.0.0          10.1.0.0.0
FRA            11.2.0.0.0          10.1.0.0.0
```

- 2) Using Enterprise Manager, navigate to the **DATA** disk group page.
 - a) Connect to Enterprise Manager Database Control as user **SYS** using the URL that was given during the previous lab. For example:
https://edrsr12p1.us.oracle.com:1158/em/
 - b) Click the **+ASM** link on the main Database page.
 - c) On the ASM home page, click the **Disk Groups** tab.
 - d) On the Automatic Storage Management Login page, enter **sys** in the Username field and **oracle_4U** in the Password field, and select **SYSASM** from the Connect As drop-down list. Select the **Save as Preferred Credentials** check box. Then click **Login**.
 - e) On the Disk Groups page, click the **DATA** link in the table.
- 3) Using Enterprise Manager Database Control, change the Database compatibility attribute of the **DATA** disk group. Set it to **11.2.0.0.0**.
 - a) On the Disk Group: DATA General subpage, click **Edit** in the Advanced Attributes section.
 - b) On the Edit Advanced Attributes for Disk Group: DATA page, enter **11.2.0.0.0** for the Database Compatibility. Then click **OK**.

Practice 5-1: ASM Fast Mirror Resync (continued)

- c) Navigate back to the **Disk Group: DATA General** subpage. Make sure that you see that the Database Compatibility field was updated. You should also see that disks are around 40% full.
- d) Attempt to change the Database Compatibility back to version **10.1.0.0.0**. What do you observe?
- 4) Use SQL*Plus to verify that the previous update was done correctly:
 - a) From a terminal window, connected as the user **oracle**, launch SQL*Plus and look at **V\$ASM_DISKGROUP**:

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ sqlplus / as sysasm
SQL> select name,compatibility,database_compatibility from
v$asm_diskgroup;

NAME          COMPATIBILITY   DATABASE_COMPATIBILITY
-----          -----          -----
DATA           11.2.0.0.0      11.2.0.0.0
FRA            11.2.0.0.0      10.1.0.0.0
```

- b) Exit SQL*Plus when finished.
- 5) Execute the **lab_05_01_05.sh** script from the **labs** directory to set up the environment for this practice. The script creates a new tablespace called **TBSJMW** in the **DATA** disk group using a 50 MB file. It then creates a new table called **SYSTEM.JMW** residing in this new tablespace. The script then inserts some rows in the newly created table.

```
$ cd ~/labs
$ ./lab_05_01_05.sh
ORACLE_SID = [oracle] ? The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle

SQL*Plus: Release 11.2.0.1.0 Production on Wed Jul 29 04:43:05
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options
```

Practice 5-1: ASM Fast Mirror Resync (continued)

```
SQL> SQL> SQL> SQL>
Tablespace created.

SQL> SQL> drop table jmw purge
*
ERROR at line 1:
ORA-00942: table or view does not exist

SQL> SQL>
Table created.

SQL> SQL>
1 row created.

SQL> SQL>
Commit complete.

SQL> SQL> 2 3 4 5 6 7
PL/SQL procedure successfully completed.

SQL> SQL> Disconnected from Oracle Database 11g Enterprise
Edition Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options
$
```

- 6) Offline the second disk that is part of the DATA disk group making sure that the Disk Repair Time attribute is set to 0:
 - a) Navigate back to the **Disk Group: DATA General** page. Select the second disk (**ASMDISK02**), and click **Offline**.
 - b) On the Confirmation page, change the Disk Repair Time from its default (**3.6 hours**) to **0.0** and click **Show SQL**.

```
ALTER DISKGROUP DATA OFFLINE DISK ASMDISK02 DROP AFTER 0.0 h
```

- c) Click **Return**.
- d) Navigate back to the Confirmation page. Click **Yes**.
- 7) What do you observe?
 - a) Navigate back to the **Disk Group: DATA General** page. You can see that ASMDISK02 is now offline. Refresh your browser page until you no longer see the offline disk. It will be renamed to something similar to this:
_DROPPED_0000_DATA
The Pending Operations will show 1 as the disk is being dropped. Click the 1 to view the progress of the rebalance operation.

Practice 5-1: ASM Fast Mirror Resync (continued)

b) Navigate back to the **Disk Group: DATA General** page. You should now see that all three of the remaining disks are around 54% full. This forced the lost mirrored extents to be rebalanced across surviving disks.

8) Modify some rows in the **SYSTEM.JMW** table (delete 499 rows). Is it working?

a) You can still modify the JMW table:

```
$ . oraenv
ORACLE_SID = [+ASM] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ sqlplus system
Enter password: oracle_4U

SQL> delete from system.jmw where rownum<500;

499 rows deleted.

SQL> commit;
```

9) Add the dropped ASM disk back to the DATA disk group:

a) You now need to wipe out the dropped disk before you can add it back. You must be root to do this:

```
# oracleasm listdisks
# oracleasm deletedisk ASMDISK02
# oracleasm createdisk ASMDISK02 /dev/xvdc
```

b) Navigate back to the **Disk Group: DATA General** page. Click **Add**.

c) On the Add Disks page, select **ORCL:ASMDISK02** from the Candidate Member Disks table. Set REBALANCE POWER to **11**.

d) Click **Show SQL**.

```
ALTER DISKGROUP DATA ADD DISK 'ORCL:ASMDISK02' SIZE 2304 M
REBALANCE POWER 11
```

e) Click **Return**.

f) On the Add Disks page, click **OK**.

10) What do you observe?

a) Navigate back to the **Disk Group: DATA General** page. Click the **Pending Operations 1** link to monitor the rebalancing operation.

b) You can see that a rebalance operation is going on for a while.

c) Allow the rebalance to complete. This may take several minutes.

Practice 5-1: ASM Fast Mirror Resync (continued)

11) Take the second disk, which is part of the DATA disk group, offline, making sure that the Disk Repair Time attribute is set to its default value of **3.6** hours. Modify the **SYSTEM.JMW** table again (delete another batch of 499 rows). What are your observations?

- Navigate back to the **Disk Group: DATA General** page. Select the second disk (**ASMDISK02**), and click **Offline**.
- On the Confirmation page, leave the default value of **3.6 Hours** in the Disk Repair Time field, and click **Yes**.
- Navigate back to the **Disk Group: DATA General** page. You can see that **ASMDISK02** is not empty. Even if you refresh your browser page, no rebalance is taking place.
- You can still modify the **SYSTEM.JMW** table.

```
$ . oraenv
ORACLE_SID = [+ASM] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ sqlplus system
Enter password: oracle_4U

SQL> delete from system.jmw where rownum<500;

SQL> commit;
```

12) Now, how would you add the offlined disk back into the DATA disk group? It is not necessary to wipe out the dropped disk.

- Navigate back to the **Disk Group: DATA General** page. Select the offline disk and click **Online**.
- On the Confirmation page, click **Yes**.
- Navigate back to the **Disk Group: DATA General** page. You should see the disk back to its level (around 41% full), without the need of any rebalance operation. The disk is added back immediately.

Practice 5-2: Using ASMCMD

In this practice, you use ASMCMD commands to manage diskgroups.

- 1) Start ASMCMD and view the contents of the **+DATA** diskgroup. Get a listing of the **DATAFILE** directory.

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ asmcmd
ASMCMD> ls +DATA/*
+DATA/ASM/:
ASMPARAMETERFILE/

+DATA/ORCL/:
CONTROLFILE/
DATAFILE/
ONLINELOG/
PARAMETERFILE/
TEMPFILE/
Spfileorcl.ora

ASMCMD> ls +DATA/ORCL/DATAFILE
EXAMPLE.260.630800437
SYSAUX.257.628766309
SYSTEM.256.628766309
TBSJMW.269.628767357
UNDOTBS1.258.628766309
USERS.259.628766309
```

- 2) Using ASMCMD, generate a list of all the commands that are allowed with the **help** command.

```
ASMCMD> help
```

- 3) Navigate to the **CONTROLFILE** directory of the **ORCL** database in the **DATA** disk group and use ASMCMD to copy the current control file to the **/tmp** directory. Use the **help cp** command for syntax guidance.

```
ASMCMD> cd +DATA/ORCL/CONTROLFILE
ASMCMD> ls
Current.260.692183799
ASMCMD> help cp
ASMCMD> cp Current.260.692183799 /tmp
copying +DATA/ORCL/CONTROLFILE/Current.260.692183799 ->
/tmp/Current.260.692183799
```

Practice 5-2: Using ASMCMD (continued)

4) Attempt to remove the current control file. Use the **help rm** command for guidance on syntax.

Note: It is important that the ORCL database instance is currently running and the **DATA** disk group is mounted.

```
ASMCMD> help rm
ASMCMD> rm Current.260.692183799
ORA-15032: not all alterations performed
ORA-15028: ASM file
'+DATA/ORCL/CONTROLFILE/Current.260.692183799' not dropped;
currently being accessed (DBD ERROR: OCISstmtExecute)
```

5) Determine the syntax for the **lsdg** command, and generate a list of all disk groups.

```
ASMCMD> help lsdg
ASMCMD> lsdg
State      Type      Rebal      Sector      Block          AU      Total_MB
Free_MB   Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED    NORMAL    N          512      4096  1048576      13824
10269          600          4834          0
N  DATA/
MOUNTED    EXTERN    N          512      4096  1048576      9216
8982          0          8982          0
N  FRA/
```

6) Determine the syntax for the **mkdg** command, and create a new disk group named **DATA2** of type **external** redundancy, using two disks: **ORCL:ASMDISK11** and **ORCL:ASMDISK12**. Verify the disk group created successfully.

```
ASMCMD> help mkdg
ASMCMD> mkdg <dg name="DATA2" redundancy="external"> <dsk
string="ORCL:ASMDISK11" /> <dsk string="ORCL:ASMDISK12" />
</dg>
ASMCMD> lsdg
State      Type      Rebal      Sector      Block          AU      Total_MB
Free_MB   Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED    NORMAL    N          512      4096  1048576      13824
10269          600          4834          0
N  DATA/
MOUNTED    EXTERN    N          512      4096  1048576      4608
4556          0          4556          0
N  DATA2/
MOUNTED    EXTERN    N          512      4096  1048576      9216
8982          0          8982          0
N  FRA/
```

Practice 5-2: Using ASMCMD (continued)

7) Determine the syntax for the `dropdg` command, and drop the **DATA2** disk group created in the last step. Verify the result.

```
ASMCMD> help dropdg
ASMCMD> dropdg DATA2
ASMCMD> lsdg
State      Type      Rebal  Sector  Block      AU  Total_MB
Free_MB  Req_mir_free_MB  Usable_file_MB  Offline_disks
Voting_files  Name
MOUNTED  NORMAL  N          512    4096  1048576    13824
10269          600          4834          0
N  DATA/
MOUNTED  EXTERN  N          512    4096  1048576    9216
8982          0          8982          0
N  FRA/
```

Practices for Lesson 6

Background: In this practice you configure connectivity between your machine and a database on one of your classmate's machines. You also configure and test an additional listener. This practice is entirely for educational purposes and no future practices rely on successful completion of this practice.

Practice 6-1: Configuring the Oracle Network to Access Another Database

Configure your network environment so that you can connect to a partner's `orcl` database. Use local naming and create a new network service name called `testorcl` that maps to your partner's `orcl` database. Test your network changes by attempting to connect to your partner's database using the `testorcl` service name.

- 1) Make a copy of your `tnsnames.ora` file. It is in your database `$ORACLE_HOME/network/admin` directory.
 - a) In a terminal window use `oraenv` to set your environment to your database home.

```
$ . oraenv
ORACLE_SID = [orcl] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$
```

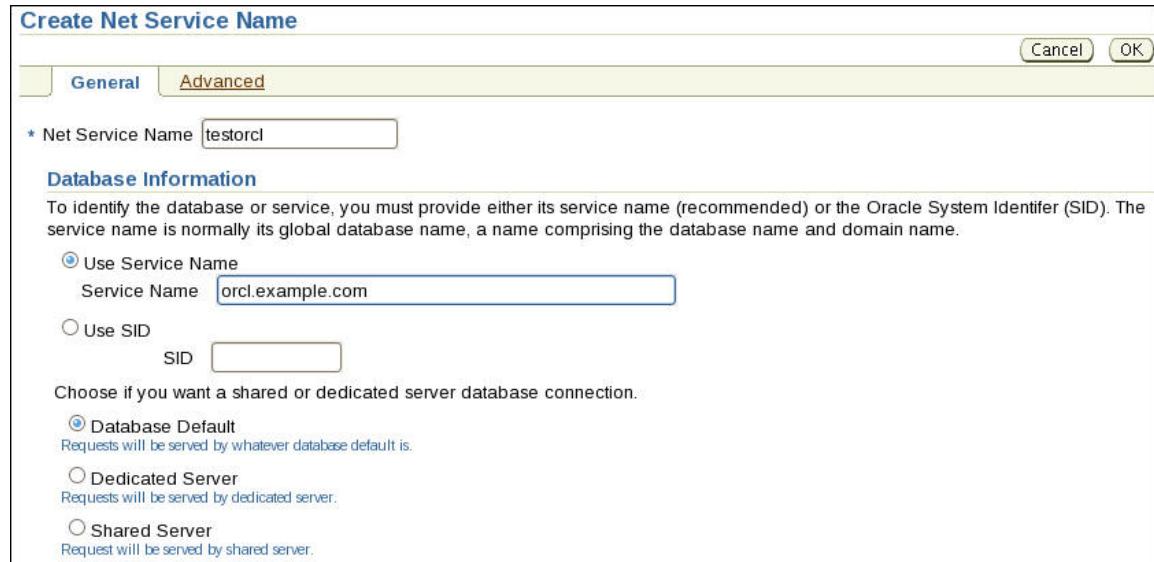
- b) Enter `cd $ORACLE_HOME/network/admin` to navigate to the `/u01/app/oracle/product/11.1.0/db_1/network/admin` directory.
 - c) Enter `cp tnsnames.ora tnsnames.old` to create a copy of the `tnsnames.ora` file.
 - d) Enter `ls -l`, if you want to see the copy and its privileges in your directory.
- 2) Navigate to the **Net Services Administration** page. Start by clicking the **Listener** link on the Database home page.
 - a) Invoke Enterprise Manager as the **SYS** user in the **SYSDBA** role for your `orcl` database.
 - b) On the Database Instance – Home page, click the **Listener** link in the **General** region.
 - c) In the **Related Links** region, click **Net Services Administration**.
- 3) Modify your local Names Resolution file so that you can connect to another database. Name the connection to a partner's `orcl` database `testorcl`.
 - a) On the **Net Services Administration** page, select **Local Naming** from the Administer drop-down list, and then click **Go**.
 - b) The Netservices Administration: Host Login page appears. If you previously saved the `oracle` username and `oracle` password as preferred credentials for your host login, they appear on the screen. If not, enter `oracle` as Username and Password, select the **Save as Preferred Credential** check box, and then click **Login**.
 - c) On the **Local Naming** page, click **Create** to enter a new network service name.
 - d) Enter `testorcl` as Net Service Name.

Practice 6-1: Configuring the Oracle Network to Access Another Database (continued)

e) Select **Use Service Name**, and enter `orcl.example.com` as Service Name.

Note: You can also choose to enter a SID by selecting the Use SID option. In this case, you must enter `orcl`.

f) Select **Database Default**.



Create Net Service Name

General **Advanced** **Cancel** **OK**

* Net Service Name `testorcl`

Database Information

To identify the database or service, you must provide either its service name (recommended) or the Oracle System Identifier (SID). The service name is normally its global database name, a name comprising the database name and domain name.

Use Service Name
Service Name `orcl.example.com`

Use SID
SID

Choose if you want a shared or dedicated server database connection.

Database Default
Requests will be served by whatever database default is.

Dedicated Server
Requests will be served by dedicated server.

Shared Server
Request will be served by shared server.

g) Click **Add** in the **Addresses** region.

h) On the Add Address page, specify the following values:

Option	Value
Protocol	TCP/IP
Port	1521
Host	<Your partner's host name or IP address—for example, edrsr25p1.us.oracle.com or 139.185.35.125 >



Add Address

Protocol **TCP/IP** **Cancel** **OK**

* Port **1521**

* Host **edrsr25p1.us.oracle.com**
The host name or IP address of the computer.

Advanced Parameters

The following parameters are introduced in Oracle version 10g.

Total Send Buffer Size
(Bytes) Cumulative size for all send operations.

Total Receive Buffer Size (Bytes)
Cumulative size for all receive operations.

i) Click **OK** to return to the Create Net Service Name properties page.

j) Click **OK**.

The Creation Message appears: Net Service “testorcl” created successfully.

Practice 6-1: Configuring the Oracle Network to Access Another Database (continued)

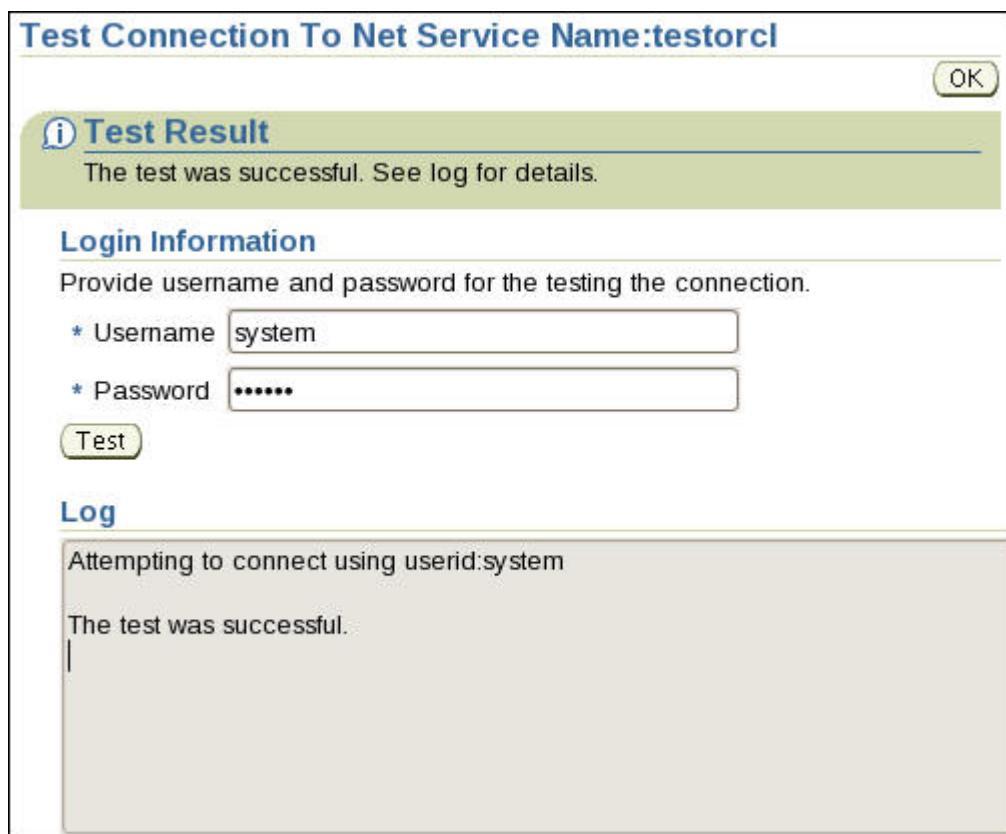
4) In Enterprise Manager, test access to your partner's `orcl` database as the `system` user with the `oracle_4U` password by using the `testorcl` Local Naming.

a) Select `testorcl` on the Local Naming page, and then click **Test Connection**.

The message "Test Connection To Net Service Name: testorcl" appears.

b) Enter `system` as Username and `oracle_4U` as Password, and then click **Test**.

The Processing page displays status information. It is followed by a success message. *If you receive any errors or warnings, resolve them.*



Click **OK** after the test is completed.

5) Test your changes to the network configuration using SQL*Plus. Enter `system@testorcl` and then enter `oracle_4U` when prompted for the password. To see your partner's information, select the `instance_name` and `host_name` columns from the `v$instance` table.

a) Ensure your environment is set for the `orcl` database by running `oraenv`.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$
```

Practice 6-1: Configuring the Oracle Network to Access Another Database (continued)

b) In a terminal window, enter:

```
$ sqlplus system@testorcl

SQL*Plus: Release 11.2.0.1.0 Production on Fri Jul 10 11:07:11
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Enter password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options

SQL>
```

The Oracle SQL*Plus window opens. *If you receive any errors or warnings, resolve them.*

c) At the SQL> prompt, enter the following command:

```
SQL> select instance_name, host_name from v$instance;

INSTANCE_NAME
-----
HOST_NAME
-----
orcl
edrsr25p1.us.oracle.com
```

Practice 6-2: Creating an Alternate Listener

In this practice, you create a second listener, called LISTENER2, using Enterprise Manager.

- 1) Create a new listener called LISTENER2. Use port 1561 for this listener.
 - a) Log in to Enterprise Manager as the **SYS** user in the **SYSDBA** role. On the Database Instance – Home page, click the **Listener** link in the **General** region.
 - b) In the **Related Links** region, click **Net Services Administration**.
 - c) On the Net Services Administration page, select **Listeners** from the **Administer** drop-down list, and then click **Go**. Enter host credentials as **oracle** and **oracle** for username and password, and then click **Login**.
 - d) On the Listeners page, which gives you an overview of the existing listeners, click the **Create** button.
The Create Listener page appears.
 - e) Enter **LISTENER2** as Listener Name, and then click **Add** to add a listener address.
 - f) Enter or confirm the following values:

Option	Value
Protocol	TCP/IP
Port	1561
Host	<Your computer's host name—for example, edrsr12p1.us.oracle.com

- g) Click **OK**.
- h) Click the **Static Database Registration** tab.



The screenshot shows the 'Create Listener' dialog box. The 'Static Database Registration' tab is selected and highlighted with a red box. The 'Listener Name' field contains 'LISTENER2'. The 'Addresses' section shows a table with one row: 'Protocol' is 'TCP/IP', 'Host' is 'edrsr12p1.us.oracle.com', and 'Port' is '1561'. The 'Add' button is visible in the top right of the 'Addresses' section.

- i) Click the **Add** button to connect the new listener with your **orcl** database.
- j) Enter the following values:

Option	Value
Service Name	orcl
Oracle Home Directory	/u01/app/oracle/product/11.2.0/dbhome_1
Oracle System Identifier	orcl

Practice 6-2: Creating an Alternate Listener (continued)

Option	Value
(SID)	

Add Database Service

Service Name: orcl

Oracle Home Directory: /u01/app/oracle/product/11.1.0/db_1

Oracle System Identifier (SID): orcl

Cancel OK

- k) Click **OK** to add the database service.
- l) Click **OK** to create the LISTENER2 listener.

Creation Message

Listener "LISTENER2" created successfully.

Listeners: /u01/app/oracle/product/11.1.0/db_1/network/admin

A listener process is identified by the listening end-points ('Host' and 'Port'), along with the other parameters like, logging and tracing levels, log/trace directories etc. All these parameters are defined in the "Listener Parameter File" (listener.ora). This page shows the status of a listener as "Started" only when the listener is running, and has been started using the "Listener Parameter File" at the same location as shown above.

Listener Name: Go Create

Edit Delete Actions Start/Stop Go

Select	Name	Protocol Details	Status	Enterprise Manager Target
	LISTENER2	Protocol: TCP/IP Host: edrsr12p1.us.oracle.com Port: 1561	Stopped	Not a target

Practice 6-2: Creating an Alternate Listener (continued)

- 2) Start the **LISTENER2** listener.
 - a) Confirm that the **LISTENER2** listener and **Start/Stop** Actions are selected, and then click **Go**.
 - b) Click **OK** on the Start/Stop page.
A confirmation message appears with a **View Details** link.
 - c) Optionally, click the **View Details** link, review the listener status information, and use the **Back** icon of your browser to return to the previous page.
- 3) Check the status of the new listener and test the new listener.
 - a) Ensure your environment is set for the **orcl** database by running **oraenv**.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$
```

- b) Issue the following commands at the command prompt

```
$ lsnrctl status LISTENER2
```

- c) Connect to your database using the new listener using an easy connect string.
Note: This method of connecting is not a recommended approach for a production environment; it is being used in this simple classroom environment just to prove the newly created listener works.

```
$ sqlplus hr/oracle_4U@your_ip_address:1561/orcl
```

Your connection is through your newly created listener. Exit SQL*Plus after you complete this step.

- 4) You can now stop this new LISTENER2 because you do not need it for the remainder of the course.

```
$ lsnrctl stop LISTENER2
```

Practices for Lesson 7

Background: You need to view existing storage structure information and create a new tablespace for the INVENTORY application. You also need to create a database user to perform all administrative tasks without using the SYS and SYSTEM accounts.

Note: Because the creation of users has not been covered, a script is provided for this practice.

Practice 7-1: Viewing Database Storage Structure Information

In this practice, you examine existing storage structure information for your database. Before you begin, you run a script that creates a new user DBA1 that will be used for your DBA tasks from now on. You must configure this user in Enterprise Manager to be one of the Administrators.

- 1) Run the `lab_07_01_01.sh` script that creates a directory that will be used later and the DBA1 user. It is located at `/home/oracle/labs`. The password for DBA1 is `oracle_4U`.
 - a) If you do not have a terminal window open from previous practices, open one now by right-clicking on your desktop and selecting **Open Terminal**.
 - b) Ensure your environment variables are set appropriately so that you can connect to your `orcl` database.

```
$ . oraenv
ORACLE_SID = [orcl] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
```

- c) Change the current directory to the `labs` directory by entering:

```
$ cd labs
```

- d) Enter the following command to run the script that creates the DBA1 user:

```
$ ./lab_07_01_01.sh
```

- e) Leave the terminal window open. You will use it again later.
- 2) Use the Setup link in the top-right corner of Enterprise Manager (EM) to define the **DBA1** user as one who can perform administrative tasks in EM. When the non-SYS user is configured, log out as the **SYS** user and log in as the **DBA1** user. Use the DBA1 user to perform the rest of these tasks, unless otherwise indicated.
 - a) In the far top-right corner of the EM window, click **Setup** and then on the Setup page select **Administrators**.



The screenshot shows the Oracle Enterprise Manager 11g interface. The top navigation bar includes links for **Setup**, **Preferences**, **Help**, **Logout**, and **Database**. The main menu bar has options for **Enterprise Manager Configuration**, **Management Services and Repository**, and **Agents**. The left sidebar has a blue header "Overview of Setup" and three links: **Administrators** (which is highlighted with a red box), **Notification**, and **Methods**. The main content area has a sub-header "Overview of Setup" with a brief description of what it does. A note at the bottom says "Enterprise Manager Configuration: lets you perform administrative operations such as".

- b) Click **Create** to add the DBA1 user to the Administrators list. This will enable the DBA1 user to perform management tasks by using Enterprise Manager.

Practice 7-1: Viewing Database Storage Structure Information (continued)



Administrators

Administrators are database users who can login to Enterprise Manager to perform management tasks like set Blackouts, email notification schedules.

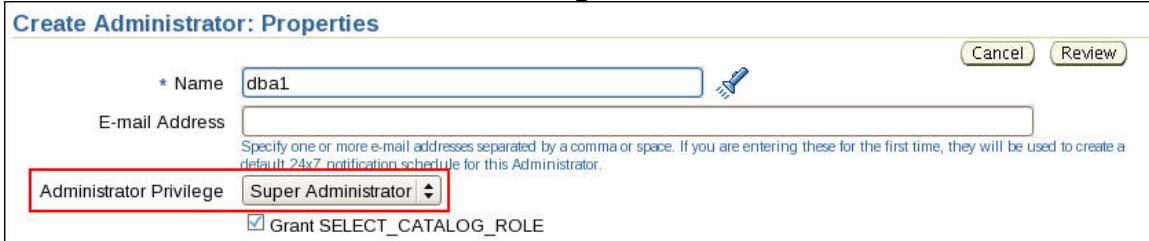
Page Refreshed Jul 8, 2009 11:38:17 Refresh AM GMT+07:00

Search Go

View Edit Subscribe to Rules Delete Create

Select Name	Access
<input checked="" type="radio"/> SYS	Super Administrator
<input type="radio"/> SYSMAN	Repository Owner
<input type="radio"/> SYSTEM	Super Administrator

c) Enter **dba1** as Name and leave Email Address blank. Select **Super Administrator** for the Administrator Privilege and then click **Review**.



Create Administrator: Properties

* Name

E-mail Address

Specify one or more e-mail addresses separated by a comma or space. If you are entering these for the first time, they will be used to create a default 24x7 notification schedule for this Administrator.

Administrator Privilege Grant SELECT_CATALOG_ROLE

Cancel Review

d) On the Create Administrator dba1: Review page, click **Finish**.

e) Now that there is a non-SYS user, click **Logout** in the top-right corner, and then click **Login**.

f) Enter **dba1** as User Name and **oracle_4U** as Password, and select **SYSDBA** as Connect As. Then click **Login**.

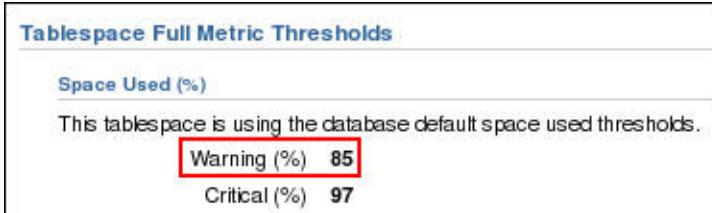
The Database Home page appears.

3) Using Enterprise Manager, view information about the **EXAMPLE** tablespace. Answer the following questions about it:

a) *Question 1:* What percentage of free space can be used up before the Warning threshold is reached?

- In Enterprise Manager, select **Server > Storage > Tablespaces**.
- Click the **EXAMPLE** tablespace name.

Answer: 85%



Tablespace Full Metric Thresholds

Space Used (%)

This tablespace is using the database default space used thresholds.

Warning (%) 85

Critical (%) 97

b) *Question 2:* How many segments are there in the EXAMPLE tablespace?

- From the **Actions** drop-down list, select **Show Tablespace Contents**, and then click **Go**.

Practice 7-1: Viewing Database Storage Structure Information (continued)

ii) The Show Tablespace Contents page appears.

Answer: 420 (Your answer may vary.)

Segments

Search

Segment Name	Type	Minimum Size (KB)	Minimum Extents
SH.CUSTOMERS	TABLE	12,288	27
PM.SYS_LOB0000073976C00034\$\$	LOBSEGMENT	5,120	20
SH.SUPPLEMENTARY_DEMOGRAPHICS	TABLE	4,096	19
PM.SYS_LOB0000073976C00054\$\$	LOBSEGMENT	4,096	19
OE.PRODUCT_DESCRIPTIONS	TABLE	3,072	18
SH.SALES.SALES_Q4_2001	TABLE PARTITION	2,048	17
SH.SALES.SALES_Q1_1999	TABLE PARTITION	1,024	16
SH.SALES.SALES_Q3_2001	TABLE PARTITION	1,024	16
SH.CUSTOMERS_PK	INDEX	1,024	16
SH.SALES.SALES_Q3_1999	TABLE PARTITION	960	15

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Previous 1-10 of 420 Next 10

c) *Question 3:* Which index in the EXAMPLE tablespace takes up the most space?

- Select INDEX from the Type drop-down list in the Search region, and then click Go.
- Notice the Size column is the sort column and that it is sorted in descending order.

Segments

Search

Segment Name	Type	Minimum Size (KB)	Minimum Extents
SH.CUSTOMERS_PK	INDEX	1,024	16
OE.PROD_NAME_IX	INDEX	512	8
OE.PRD_DESC_PK	INDEX	320	5
SH.CUSTOMERS_YOB_BIX	INDEX	192	3
SH.TIMES_PK	INDEX	128	2

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Previous 1-10 of 78 Next 10

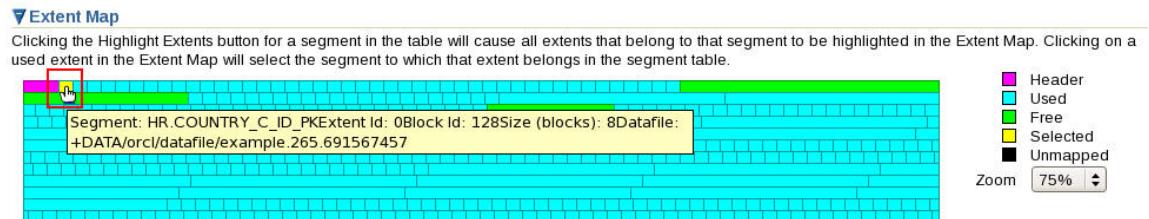
Answer: SH.CUSTOMERS_PK

d) *Question 4:* Which segment is stored physically first in the tablespace? That is, which one is stored right after the tablespace header?

- Scroll to the bottom of the page, and then click the plus icon to the left of the Extent Map label.

Practice 7-1: Viewing Database Storage Structure Information (continued)

- ii) After several seconds, the extent map appears. Note that the map legend indicates that pink is the tablespace header.
- iii) Scroll back to the top of the page, select **All Types** from the **Type** drop-down list, and then click **Go**.
- iv) Click the extent just to the right of the tablespace header extent (the extent will turn yellow to show that it is selected). Notice that if you move the cursor over the segment, it tells you the name of the segment stored in that location.



- v) Scroll to the top of the page again, and note the segment that is being pointed to:

Highlight Extents	Segment Name	Type	Size (KB)	Extents
<input type="radio"/>	SH.COSTS.COSTS_Q1_2003	TABLE PARTITION	64	1
<input type="radio"/>	SH.COSTS.COSTS_Q2_2003	TABLE PARTITION	64	1
<input type="radio"/>	SH.COSTS.COSTS_Q3_2003	TABLE PARTITION	64	1
<input type="radio"/>	SH.COSTS.COSTS_Q4_2003	TABLE PARTITION	64	1
<input type="radio"/>	HR.REG_ID_PK	INDEX	64	1
<input checked="" type="radio"/>	HR.COUNTRY_C_ID_PK	INDEX	64	1
<input type="radio"/>	HR.LOC_ID_PK	INDEX	64	1
<input type="radio"/>	HR.LOC_CITY_IX	INDEX	64	1
<input type="radio"/>	HR.LOC_STATE_PROVINCE_IX	INDEX	64	1
<input type="radio"/>	HR.LOC_COUNTRY_IX	INDEX	64	1

Answer: HR.COUNTRY_C_ID_PK

Practice 7-2: Creating a Tablespace

In this practice, you create the Inventory tablespace that will be used in a later practice.

- 1) Create a new, locally managed tablespace (LMT) called **INVENTORY** of size **5 MB**.
 - a) In Enterprise Manager, select **Server**> **Storage** > **Tablespaces**.
 - b) Click **Create**.
 - c) Enter **INVENTORY** as the tablespace name, and verify that Extent Management is **Locally Managed**, Type is **Permanent**, Status is **Read Write**, and Use bigfile tablespace is not selected.
 - d) Click **Add** in the Datafiles region.
 - e) On the Add Datafile page, select **Automatic Storage Management** for Storage Type, ensure that **DATA** is selected for DiskGroup, and enter **5 MB** as File Size. Then click **Continue**.

The screenshot shows the 'Add Datafile' dialog box. The 'Storage Type' is set to 'Automatic Storage Management'. The 'DiskGroup' is set to 'DATA'. The 'Template' is set to '<Default>'. The 'Tablespace' is set to 'INVENTORY'. The 'File Size' is set to '5 MB'. The 'Reuse Existing File' checkbox is unchecked. The 'Storage' tab is selected, showing options for 'Automatically extend datafile when full (AUTOEXTEND)', 'Increment' (set to '0 KB'), and 'Maximum File Size' (set to 'Unlimited'). The 'Reuse Existing File' checkbox is checked. The 'Cancel' and 'Continue' buttons are visible at the top right.

- f) Click the **Storage** tab, and verify that Extent Allocation is **Automatic**, Segment Space Management is **Automatic**, Compression Options is **Disabled**, and Logging is set to **Yes**.

Practice 7-2: Creating a Tablespace (continued)

g) Click the **General** tab and review your settings.

Create Tablespace

Information
Modification to the datafile will not take effect until you click "OK" button.

General **Storage**

* Name **INVENTORY**

Extent Management **Type** **Status**

Locally Managed Permanent Read Write

Dictionary Managed Set as default permanent tablespace Read Only

Undo Encryption [Encryption Options](#) Offline

Temporary Set as default temporary tablespace

Undo Retention Guarantee Yes No

Datafiles

Use bigfile tablespace
Tablespace can have only one datafile with no practical size limit.

Add

Select Name	Directory	Size (MB)
<input checked="" type="radio"/> <ASM Generated>	+DATA/	5.00

h) Click **Show SQL** to see the SQL that will be run, and then click **Return**

i) Click **OK**, and a successful Update Message appears.

2) As the **DBA1** user, run the **lab_07_02_02.sql** script to create and populate a table (called X) in the **INVENTORY** tablespace. What error do you eventually see?

a) In a terminal window, navigate to the **labs** directory. Remember to use **oraenv** to set your environment for the **orcl** database if you have not done so already.

```
$ cd labs
```

b) Log in to SQL*Plus as the **dba1** user (with a password of **oracle_4U**) and run the **lab_07_02_02.sql** script.

Note: Remember to use **oraenv** to set your environment to the **orcl** database, if you have not already done so in your terminal window.

```
$ sqlplus dba1
```

```
SQL*Plus: Release 11.2.0.1.0 Production on Wed Jul 8 12:06:50  
2009
```

Practice 7-2: Creating a Tablespace (continued)

```
Copyright (c) 1982, 2009, Oracle. All rights reserved.
```

```
Enter password:
```

```
Connected to:
```

```
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -  
Production  
With the Partitioning, Automatic Storage Management, OLAP,  
Data Mining  
and Real Application Testing options
```

```
SQL> @lab_07_02_02.sql
```

c) Note that there is eventually an error ORA-01653 stating that the table cannot be extended. There is not enough space to accommodate all the rows to be inserted.

```
...  
SQL> insert into x select * from x  
  2 /  
  
1024 rows created.  
  
SQL> insert into x select * from x  
  2 /  
insert into x select * from x  
*  
ERROR at line 1:  
ORA-01653: unable to extend table DBA1.X by 128 in tablespace  
INVENTORY
```

```
SQL> commit  
  2 /  
  
Commit complete.  
  
SQL> quit  
Disconnected from Oracle Database 11g Enterprise Edition  
Release 11.1.0.6.0 - Production  
With the Partitioning, OLAP, Data Mining and Real Application  
Testing options
```

3) Go to the Enterprise Manager window and increase the amount of space available for the **INVENTORY** tablespace. For educational purposes, you will accomplish this using two different methods. First, increase the size of the current datafile to **40 MB**. Then, to show that both ASM and non-ASM datafiles can exist for the same tablespace, add a second datafile using file system storage. This second datafile should be **30 MB** in size. For both techniques use the show SQL functionality to view the supporting SQL statements.

- Select **Server> Storage > Tablespaces**.
- Select the **INVENTORY** tablespace, and then click **Edit**.

Practice 7-2: Creating a Tablespace (continued)

- c) In the Datafiles region, click **Edit**.
- d) Change File Size from 5 MB to **40 MB**.
- e) Click **Continue** to return to the General tabbed page.
- f) Click **Show SQL** to see the SQL that will be run. Note that it is an **ALTER DATABASE** statement. Click **Return**.

Show SQL

ALTER DATABASE DATAFILE '+DATA/orcl/datafile/inventory.268.693647553' RESIZE 40M

Return

- g) In the Datafiles region, click **Add**.
- h) Select **File System** for the Storage Type. Enter **inventory02.dbf** for the File Name. Enter **/u01/app/oracle/oradata/orcl** for the File Directory. Enter **30 MB** for the File Size.
Note: This directory was created by the script you ran earlier.
- i) Click **Continue** to return to the General tabbed page.
- j) Click **Show SQL** to see the SQL that will be run. Note that it is an **ALTER DATABASE** statement. Click **Return**.

Show SQL

ALTER TABLESPACE "INVENTORY" ADD DATAFILE '/u01/app/oracle/oradata/orcl/inventory02.dbf' SIZE 30M

Return

- k) Click **Apply**.
- l) Notice now that there are now two datafiles for the INVENTORY tablespace, one that is using ASM storage and the other using file system (non-ASM) storage.

Datafiles				
Select		Name	Directory	Size (MB)
<input checked="" type="radio"/>		inventory02.dbf	/u01/app/oracle/oradata/orcl/	30.00
<input type="radio"/>		inventory.268.693647553	+DATA/orcl/datafile/	40.00

- 4) Go back to the terminal window and run the **lab_07_02_04.sql** script. It drops the table and re-executes the original script that previously returned the space error.
 - a) Go to the terminal window.
 - b) Log in to SQL*Plus as the **dba1** user (with a password of **oracle_4U**) and run the **lab_07_02_04.sql** script.
Note: Remember to use oraenv to set your environment to the orcl database if you have not already done so in your terminal window.

```
$ sqlplus dba1

SQL*Plus: Release 11.2.0.1.0 Production on Wed Jul 8 12:06:50
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.
```

Practice 7-2: Creating a Tablespace (continued)

```
Enter password:
```

```
Connected to:
```

```
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -  
Production  
With the Partitioning, Automatic Storage Management, OLAP,  
Data Mining  
and Real Application Testing options
```

```
SQL> @lab_07_02_04.sql
```

- c) Note that the same number of row inserts are attempted, and there is no error because of the increased size of the tablespace.
- 5) In a terminal window, run the **lab_07_02_05.sql** script in SQL*Plus as the **dba1** user to clean up the tablespace for later practice sessions.
Note: Remember to use `oraenv` to set your environment to the `orcl` database, if you have not already done so in your terminal window.

```
$ sqlplus dba1
```

```
SQL*Plus: Release 11.2.0.1.0 Production on Wed Jul 8 12:06:50  
2009
```

```
Copyright (c) 1982, 2009, Oracle. All rights reserved.
```

```
Enter password:
```

```
Connected to:
```

```
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -  
Production  
With the Partitioning, Automatic Storage Management, OLAP,  
Data Mining  
and Real Application Testing options
```

```
SQL> @lab_07_02_05.sql
```

Practices for Lesson 8

Background: You need to create a user account for Jenny Goodman, the new human resources department manager. There are also two new clerks in the human resources department, David Hamby and Rachel Pandya. All three must be able to log in to the `orcl` database and to select data from, and update records in, the `HR.EMPLOYEES` table. The manager also needs to be able to insert and delete new employee records. Ensure that if the new users forget to log out at the end of the day, they are automatically logged out after 15 minutes. You also need to create a new user account for the inventory application that you are installing.

Practice 8-1: Creating and Using a Profile

In this practice, you create the INVENTORY user to own the new Inventory application. You create a profile to limit the idle time of users. If a user is idle or forgets to log out after 15 minutes, the user session is ended.

- 1) **Mandatory task:** Review and run the `lab_08_01_01.sh` script (located in the `/home/oracle/labs` directory) to create the **INVENTORY** user (with a password of `oracle_4U`), which you will use in the next practice.

- a) In a terminal window enter:

```
$ cd $HOME/labs
$ cat lab_08_01_01.sh
# Oracle Database 11g: Administration Workshop I
# Oracle Server Technologies - Curriculum Development
#
# ***Training purposes only***
#***Not appropriate for production use***
#
# Start this script as OS user: oracle
#   This script creates the INVENTORY schema user
#   The DROP command fails the first time
#   you execute the script.
#   The error can be ignored.

cd ~/labs

. set_db.sh

sqlplus / as sysdba << EOF
drop user inventory cascade;

create user inventory identified by oracle_4U
default tablespace inventory;

grant connect, resource to inventory;

exit;
EOF
$ ./lab_08_01_01.sh

SQL*Plus: Release 11.2.0.1.0 Production on Wed Jul 8 13:00:36
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
```

Practice 8-1: Creating and Using a Profile (continued)

```
and Real Application Testing options

SQL> SQL> drop user inventory cascade
      *
ERROR at line 1:
ORA-01918: user 'INVENTORY' does not exist

SQL> SQL> 2
User created.

SQL> SQL>
Grant succeeded.

SQL> SQL> Disconnected from Oracle Database 11g Enterprise
Edition Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options
$
```

- 2) Create a profile named **HRPROFILE** that allows only 15 minutes idle time.
 - a) Invoke Enterprise Manager as the **DBA1** user in the **SYSDBA** role for your **orcl** database.
 - b) Click the **Server** tab, and then click **Profiles** in the Security section.
 - c) Click the **Create** button.
 - d) Enter **HRPROFILE** in the Name field.
 - e) Enter **15** in the Idle Time (Minutes) field.
 - f) Leave all the other fields set to **DEFAULT**.
 - g) Click the **Password** tab, and review the Password options, which are currently all set to **DEFAULT**.
 - h) Optionally, click the **Show SQL** button, review your underlying SQL statement, and then click **Return**.
 - i) Finally, click **OK** to create your profile
- 3) Set the **RESOURCE_LIMIT** initialization parameter to **TRUE** so that your profile limits are enforced.
 - a) Click the **Server** tab, and then click **Initialization Parameters** in the Database Configuration section.
 - b) Enter **resource_limit** in the Name field, and then click **Go**.
 - c) Select **TRUE** from the Value drop-down list, and then click **Apply**.

Practice 8-2: Creating Roles

In this practice, you create the **HRCLERK** and **HRMANAGER** roles that will be used in the next practice.

- 1) Create the role named **HRCLERK** with **SELECT** and **UPDATE** permissions on the **HR.EMPLOYEES** table.
 - a) Click the **Server** tab and then click **Roles** in the Security section.
 - b) Click the **Create** button.
 - c) Enter **HRCLERK** in the Name field. This role is not authenticated.
 - d) Click **Object Privileges** tab.
 - e) Select **Table** from the Select Object Type drop-down list, and then click **Add**.
 - f) Enter **HR.EMPLOYEES** in the Select Table Objects field.
 - g) Move the **SELECT** and **UPDATE** privileges to the Selected Privileges box. Click **OK**.
 - h) Click the **Show SQL** button, and review your underlying SQL statement.

Show SQL **Return**

```
CREATE ROLE "HRCLERK" NOT IDENTIFIED
GRANT SELECT ON "HR"."EMPLOYEES" TO "HRCLERK"
GRANT UPDATE ON "HR"."EMPLOYEES" TO "HRCLERK"
```

- i) Click **Return**, and then click **OK** to create the role.
- 2) Create the role named **HRMANAGER** with **INSERT** and **DELETE** permissions on the **HR.EMPLOYEES** table. Grant the **HRCLERK** role to the **HRMANAGER** role.
 - a) Click the **Server** tab, and then click **Roles** in the Security section.
 - b) Click **Create**.
 - c) Enter **HRMANAGER** in the Name field. This role is not authenticated.
 - d) Click **Object Privileges** tab.
 - e) Select **Table** from the Select Object Type drop-down list, and then click **Add**.
 - f) Enter **HR.EMPLOYEES** in the Select Table Objects field.
 - g) Move the **INSERT** and **DELETE** privileges to the Selected Privileges box. Click **OK**.
 - h) Click the **Roles** tab, and then click **Edit List**.
 - i) Move the **HRCLERK** role into the Selected Roles box, and then click **OK**.

Practice 8-2: Creating Roles (continued)

j) Click the **Show SQL** button, and review your underlying SQL statement.

Show SQL

Return

```
CREATE ROLE "HRMANAGER" NOT IDENTIFIED
GRANT DELETE ON "HR"."EMPLOYEES" TO "HRMANAGER"
GRANT INSERT ON "HR"."EMPLOYEES" TO "HRMANAGER"
GRANT "HRCLERK" TO "HRMANAGER"
```

k) Click **Return**, and then click **OK** to create the role.

Practice 8-3: Creating and Configuring Users

In this practice, you create the following users and assign appropriate profiles and roles to these users:

Name	Username	Description
David Hamby	DHAMBY	A new HR Clerk
Rachel Pandya	RPANDYA	A new HR Clerk
Jenny Goodman	JGOODMAN	A new HR Manager

- 1) Create an account for David Hamby, a new HR clerk.
 - a) Click the **Server** tab, and then click **Users** in the Security section.
 - b) Click **Create**, and enter **DHAMBY** in the Name field.
 - c) Select **HRPROFILE** for the Profile.
 - d) Select Password Authentication, and enter **newuser** as password. Enter it into the **Confirm Password** field also. Select the **Expire Password now** check box so that David will have to change the password the first time he logs in.
 - e) Click the **Roles** tab. Notice that the CONNECT role has automatically been assigned to the user.
 - f) Add the **HRCLERK** role by clicking Edit List and moving the **HRCLERK** role into the Selected Roles box. Click **OK** to close the Modify Roles window.
 - g) Click **OK** again to create the user.
- 2) Create an account for Rachel Pandya, another new HR clerk. Repeat the steps shown above in step 1 but with **RPANDYA** as the username.
- 3) Create an account for Jenny Goodman, the new HR manager. Repeat the steps under step 1 but use **JGOODMAN** as the username and select the **HRMANAGER** role instead of the **HRCLERK** role.
 - a) Click the **Show SQL** button and review your underlying SQL statement.

Show SQL **Return**

```
CREATE USER "JGOODMAN" PROFILE "HRPROFILE" IDENTIFIED BY *****
PASSWORD EXPIRE ACCOUNT UNLOCK
GRANT "CONNECT" TO "JGOODMAN"
GRANT "HRMANAGER" TO "JGOODMAN"
```

- b) Click **Return**, and then click **OK** to create the user.
- 4) Test the new users in SQL*Plus. Connect to the **orcl** database as the **DHAMBY** user. Use **oracle_4U** as the new password. Select the row with **EMPLOYEE_ID=197** from the **HR.EMPLOYEES** table. Then attempt to delete it. (You should get the “insufficient privileges” error.)
 - a) In a terminal window, enter:

```
$ . oraenv
```

Practice 8-3: Creating and Configuring Users (continued)

```
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ sqlplus dhamby
```

Or, if you are already in SQL*Plus, use the CONNECT command. If you reconnect as dhamby in SQL*Plus, the login and change-of-password session looks like this:

```
SQL> CONNECT dhamby
Enter password: newuser    <<<Password does not appear on screen
ERROR:
ORA-28001: the password has expired

Changing password for dhamby
New password: oracle_4U    <<<Password does not appear on screen
Retype new password: oracle_4U    <<<Password does not appear on screen
Password changed

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP, Data
Mining
and Real Application Testing options
SQL>
```

b) Select the salary for **EMPLOYEE_ID=197** from the **HR.EMPLOYEES** table.

```
SQL> SELECT salary FROM hr.employees WHERE EMPLOYEE_ID=197;

      SALARY
      -----
      3000
```

c) Now attempt to delete the same record from the **hr.employees** table.

```
SQL> DELETE FROM hr.employees WHERE EMPLOYEE_ID=197;
DELETE FROM hr.employees WHERE EMPLOYEE_ID=197
*
ERROR at line 1:
ORA-01031: insufficient privileges
```

5) Repeat the test as the **JGOODMAN** user. Use **oracle_4U** as the new password. After deleting the row, issue a rollback, so that you still have the original 107 rows.

a) Connect to the **orcl** database as the **JGOODMAN** user.

```
SQL> connect jgoodman
Enter password:
ERROR:
ORA-28001: the password has expired
<Change the password to oracle_4U as shown above>
```

Practice 8-3: Creating and Configuring Users (continued)

b) Select the row with EMPLOYEE_ID=197 from the HR.EMPLOYEES table.

```
SQL> SELECT salary FROM hr.employees WHERE EMPLOYEE_ID=197;  
  
SALARY  
-----  
3000
```

c) Now delete the same row from the HR.EMPLOYEES table.

```
SQL> DELETE FROM hr.employees WHERE EMPLOYEE_ID=197;  
  
1 row deleted.
```

d) Roll back the delete operation (because this was just a test).

```
SQL> rollback;  
  
Rollback complete.
```

e) Confirm that you still have 107 rows in this table.

```
SQL> SELECT COUNT(*) FROM hr.employees;  
  
COUNT(*)  
-----  
107  
  
SQL>
```

Question 1: Where was the row stored after deletion?

Answer: It was stored in the Undo tablespace.

Question 2: When you created the new users, you did not select a default or temporary tablespace. What determines the tablespaces that the new users will use?

Answer: The system-defined default permanent and temporary tablespaces

Question 3: You did not grant the CREATE SESSION system privilege to any of the new users, but they can all connect to the database. Why?

Answer: Because Enterprise Manager automatically assigns the CONNECT role to the new users, and CREATE SESSION is contained within that role

6) Use SQL*Plus to connect to the **orcl** database as the **RPANDYA** user. Change the password to **oracle_4U**. (You must change the password, because this is the first connection as RPANDYA.) Leave RPANDYA connected during the next lesson or at the end of the day. HRPFILE specifies that users whose sessions are inactive for more than 15 minutes will automatically be logged out. Verify that the user was automatically logged out by trying to select from the **HR.EMPLOYEES** table again.

Practice 8-3: Creating and Configuring Users (continued)

```
ERROR at line 1:  
ORA-02396: exceeded maximum idle time, please connect again
```

Practices for Lesson 9

Background: The Help desk just received a call from Susan Mavris, an HR representative, complaining that the database is “frozen.” Upon questioning the user, you find that she was trying to update John Chen’s personnel record with his new phone number, but when she entered the new data, her session froze and she could not do anything else. SQL script files are provided for you in the `/home/oracle/labs` directory.

Practice 9-1: Resolving Lock Conflicts

In this practice, you use two separate SQL*Plus sessions to cause a lock conflict. Using Enterprise Manager, you detect the cause of the lock conflict and then resolve the conflict. For your convenience, the SQL code that will cause the lock conflict has been provided in scripts that you run during this practice.

- 1) Make an uncommitted update to the row in question by running the **lab_09_01_01.sql** script. This script first creates the users (smavris and ngreenberg) that are involved in this practice and the hremployee role that will give these new users access to the hr.employee table. It then logs in to SQL*Plus as the ngreenberg user and performs an update on the hr.employee table. The script does not perform a commit, leaving the update uncommitted in this session.
 - a) Ensure your environment is configured for the **orcl** database by running **oraenv**.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$
```

- b) Enter the following to run the script. When the script completes executing, you will see a note stating that an uncommitted update has been made.

```
$ sqlplus dba1

SQL*Plus: Release 11.2.0.1.0 Production on Thu Jul  9
03:57:42 2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Enter password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options

SQL> @lab_09_01_01.sql
Creating users...
...
... Some output not shown here to conserve space ...
Connecting as ngreenberg ...
Connected.
SQL> show user
USER is "NGREENBERG"
```

Practice 9-1: Resolving Lock Conflicts (continued)

```
SQL> update hr.employees set phone_number='650.555.1212'
  where employee_id = 110 ;

1 row updated.

SQL> prompt User "ngreenberg" made an update and left it
uncommitted in this session.
User "ngreenberg" made an update and left it uncommitted in
this session.
SQL>
SQL>
SQL>
```

- c) Leave this session connected in the state that it is currently. **Do not** exit at this time.
- 2) Make an attempt to update the same row in a separate session by running, in a separate terminal window, the **lab_09_01_02.sql** script. Make sure that you see the message “Update is being attempted now” before moving on. Do not worry if the session seems to “hang”—this is the condition that you are trying to create.
 - a) Open a terminal window to start another command shell, and enter the following to run the second script.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ sqlplus dba

SQL*Plus: Release 11.2.0.1.0 Production on Thu Jul  9
04:04:47 2009

Copyright (c) 1982, 2009, Oracle.  All rights reserved.

Enter password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options

SQL> @lab_09_01_02.sql
Sleeping for 20 seconds to ensure first process gets the
lock first.

PL/SQL procedure successfully completed.

Sleep is finished.
Connected.
```

Practice 9-1: Resolving Lock Conflicts (continued)

USER is "SMAVRIS"
Update is being attempted now.

b) Notice that this session appears to be hung. Leave this session as is and move on to the next step.

3) Using Enterprise Manager, click the **Blocking Sessions** link on the Performance page and detect which session is causing the locking conflict.

- In Enterprise Manager, click the **Performance** page.
- Click **Blocking Sessions** in the **Additional Monitoring Links** area. You should see the following:

Blocking Sessions														
Page Refreshed Jul 9, 2009 4:13:46 AM GMT+07:00 Refresh														
		View Session		Kill Session										
Expand All Collapse All														
Select	Username	Sessions Blocked	Session ID	Serial Number	SQL ID	Wait Class	Wait Event	P1 Value	P2 Value	P3 Value	Seconds in Wait			
<input type="radio"/>	Blocking Sessions													
<input checked="" type="radio"/>	NGREENBERG	1	51	7460		Idle	SQL*Net message from client	16508152321	0	862				
<input type="radio"/>	SMAVRIS	0	52	4319 6smgtv6h8958b	Application	eng: TX - row lock contention	eng: TX - row lock contention	1415053318 327697778		383				

4) What was the last SQL statement that the blocking session executed?

- Select the **NGREENBERG** session, and then click **View Session**.
- Click the hash value link for **Previous SQL**.

Session Details: 51 (NGREENBERG)																
Collected From Target Jul 9, 2009 4:21:57 AM GMT+07:00																
View Data Real Time: 15 Second Refresh Refresh Kill Session Enable SQL Trace																
General		Activity		Statistics		Open Cursors		Blocking Tree		Wait Event History		Parallel SQL	SQL Monitoring			
Server		Client		Application												
Current Status INACTIVE		OS User Name oracle		Current SQL None												
Serial Number 7460		OS Process ID 6962		Current SQL Command UNKNOWN												
DB Process Name NGREENBERG		Host edrsr12p1.us.oracle.com		Previous SQL 6smgtv6h8958b												
OS Process ID 7129		Terminal pts/2		Last Call Duration 22:34 (mm:ss)												
Login Time Jul 9, 2009 3:59:24 AM		Current Client ID Unavailable		SQL Trace DISABLED												
Login Duration 22:34 (mm:ss)		Current Client Info Unavailable		Current SQL Trace Level 1												
Connection Type DEDICATED		Trace With Wait Information DISABLED		Trace With Bind Information DISABLED												
Type USER		Open Cursors 35		Program sqlplus@edrsr12p1.us.oracle.com (TNS V1-V3)												
Resource Consumer Group Unavailable		Service SYS\$USERS		Current Module SQL*Plus												
		Current Action Unavailable														

c) Note the SQL that was most recently run.

SQL Details: 6smgtv6h8958b												
Switch to SQL ID: <input type="text"/> Go View Data Real Time: Manual Refresh Refresh SQL Worksheet Schedule SQL Tuning Advisor SQL Repair Advisor												
Text   update hr.employees set phone_number='650.555.1212' where employee_id = 110												

5) Resolve the conflict in favor of the user who complained, by killing the *blocking* session. What SQL statement resolves the conflict?

Practice 9-1: Resolving Lock Conflicts (continued)

- a) Click the browser's **Back** button.
- b) Now, on the Session Details: NGREENBERG page, click **Kill Session**.
- c) Leave the Options set to **Kill Immediate**, and then click **Show SQL** to see the statement that is going to be executed to kill the session.

Note: Your session and serial number are most likely to be different from those shown here.

Show DDL

Return

```
ALTER SYSTEM KILL SESSION '51,7460' IMMEDIATE
```

- d) Click **Return**, and then click **Yes** to carry out the **KILL SESSION** command.
- 6) Return to the SQL*Plus command window, and note that SMAVRIS's update has now completed successfully. It may take a few seconds for the success message to appear.

```
USER is "SMAVRIS"
Update is being attempted now.

1 row updated.

Update is completed.
SQL>
```

- 7) Try issuing a SQL select statement in the NGREENBERG session. What do you see?

```
SQL> SELECT sysdate from dual;
SELECT sysdate from dual
*
ERROR at line 1:
ORA-03135: connection lost contact
Process ID: 7129
Session ID: 51 Serial number: 7460

SQL>
```

Answer: The session has been disconnected.

Close all open SQL sessions by entering **exit**, and then close the terminal windows.

Practices for Lesson 10

Background: The business users and management in your organization decide, that they need to have a 48-hour retention of undo in the Oracle database to support their flashback needs. Your task is to configure the `orcl` database to support this requirement.

Practice 10-1: Managing Undo Data

In this practice, you first view your system activity regarding undo, and then you configure the `orcl` database to support 48-hour retention for flashback operations.

- 1) In Enterprise Manager, as the **DBA1** user, view the undo related system activity.
 - a) Click the **Server** tabbed page and select **Automatic Undo Management** in the Database Configuration section.
 - b) Click the **System Activity** tabbed page.

Automatic Undo Management

In the General tab, you can view the current undo settings for your instance and use the Undo Advisor to analyze the undo tablespace requirements. This analysis can be performed based on the specified analysis period or the desired undo retention. The system activity for the specified time period can be viewed in the System Activity tab.

[General](#) [System Activity](#)

System Activity During Analysis Period

Selected Analysis Time Period	Jul 2, 2009 5:00:00 AM GMT+07:00 To Jul 9, 2009 5:00:00 AM GMT+07:00
Longest Running Query (minutes)	18.0
Average Undo Generation Rate (KB/minute)	75.0
Maximum Undo Generation Rate (KB/minute)	1,385.0
Queries failed due to low Retention	0
Transactions failed due to small Undo Tablespace	0

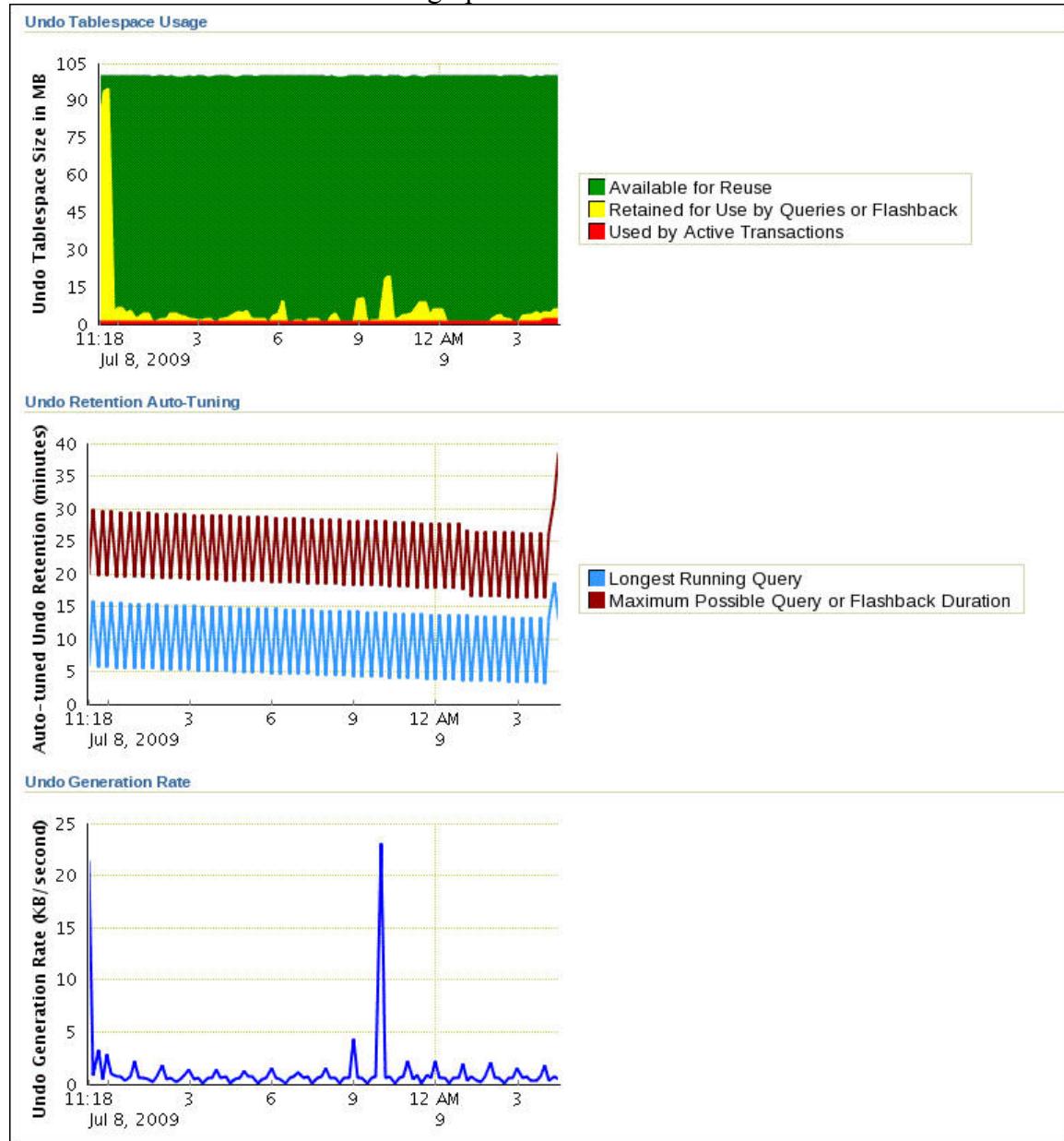
[Show Graph](#)

Note: Your information will look different on all analysis screenshots, based on your analysis period and the system activity during this period.

- c) *Question:* Looking at the preceding screenshot, how many errors did this system encounter?
Answer: None
- d) *Question:* Looking at the preceding screenshot, what is the duration of the longest running query?
Answer: 18 minutes (Your answer may be different.)

Practice 10-1: Managing Undo Data (continued)

e) Click the Plus icon to show related graphs.



f) *Question:* How many graphs are displayed?

Answer: Three. (Undo Tablespace Usage, Undo Retention Auto-Tuning, and Undo Generation Rate)

g) *Question:* Looking at the preceding Undo Retention Auto-Tuning graph, could this system support flashback above and beyond the current longest running query?

Answer: Yes, (but most likely not enough to support the required 48 hours).

2) Modify the undo retention time and calculate the undo tablespace size to support the requested 48-hour retention.

Practice 10-1: Managing Undo Data (continued)

- a) Click the **General** tab to go back to the General Automatic Undo Management page.
- b) Under the Undo Advisor section, select “**Specified manually to allow for longer duration queries or flashback.**”
- c) Enter **48 hours** as Duration and click the **Run Analysis** button.

Undo Retention Settings Undo Retention (minutes) 15 Retention Guarantee No	Undo Tablespace for this Instance Tablespace UNDOTBS1 Change Tablespace Size (MB) 100 Auto-Extensible Yes
Undo Advisor: Undo Retention and Undo Tablespace Sizing Advice Undo retention is the length of time that undo data is retained in the undo tablespaces. Undo data must be retained for the length of the longest running query, the longest running transaction, and the longest flashback duration (except for Flashback Database). The undo tablespace should be sized large enough to hold the undo generated by the database during the undo retention period. Note that the undo retention parameter is also used as the retention value for LOB columns.	
Analysis Period Analysis Time Period Last Seven Days Desired Undo Retention <input type="radio"/> Automatically chosen based on longest query in analysis period <input checked="" type="radio"/> Specified manually to allow for longer duration queries or flashback Duration 48 hours Run Analysis	
Analysis Results Edit Undo Tablespace Edit Undo Retention	

Practice 10-1: Managing Undo Data (continued)

d) When the Undo Advisor is finished, examine the results.

Automatic Undo Management

In the General tab, you can view the current undo settings for your instance and use the Undo Advisor to analyze the undo tablespace requirements. This analysis can be performed based on the specified analysis period or the desired undo retention. The system activity for the specified time period can be viewed in the System Activity tab.

General **System Activity**

Undo Retention Settings

Undo Retention (minutes)	15
Retention Guarantee	No

Undo Tablespace for this Instance

Tablespace	UNDOTBS1
Size (MB)	100
Auto-Extensible	Yes

Undo Advisor: Undo Retention and Undo Tablespace Sizing Advice

Undo retention is the length of time that undo data is retained in the undo tablespaces. Undo data must be retained for the length of the longest running query, the longest running transaction, and the longest flashback duration (except for Flashback Database). The undo tablespace should be sized large enough to hold the undo generated by the database during the undo retention period. Note that the undo retention parameter is also used as the retention value for LOB columns.

Analysis Period

Analysis Time Period: Last Seven Days

Desired Undo Retention:

Automatically chosen based on longest query in analysis period

Specified manually to allow for longer duration queries or flashback

Duration: 48 hours

Analysis Results

Selected Analysis Time Period: Jul 2, 2009 8:00:00 AM GMT+07:00 To Jul 9, 2009 8:00:00 AM GMT+07:00

Minimum Required Undo Tablespace Size (MB): 249

Recommended Undo Tablespace Size (MB): 20

TIP Recommended size is three times the minimum size to allow for workload fluctuations

Potential Problems: No Problem Found

Recommendations: No Recommendation

Show Graph

Note: Your recommended size might be different from what is shown here.

e) Click the **Show SQL** button in the upper-right corner of the General Automatic Undo Management page.

Show SQL

```
ALTER SYSTEM SET UNDO_RETENTION = 172800
```

Return

f) This command will change the undo retention to support the 48-hour requirement. Review the SQL statement and click **Return**.

g) Click **Apply** to make the change to undo retention.

h) Now adjust the undo tablespace size by clicking the **Edit Undo Tablespace** button.

i) Scroll down to Datafiles and click **Edit** to make a change to the datafile file size for the Undo tablespace.

j) Change the file size to the Minimum Required Undo Tablespace Size that was determined when you ran the Undo Advisor (249 MB is the value in the screenshot above) and click **Continue**.

Practice 10-1: Managing Undo Data (continued)

k) Verify the SQL commands that will be executed by clicking Show SQL.

Show SQL

Return

```
ALTER DATABASE DATAFILE '+DATA/orcl/datafile/undotbs1.258.691672083' RESIZE 249M
ALTER DATABASE DATAFILE '+DATA/orcl/datafile/undotbs1.258.691672083' AUTOEXTEND ON NEXT 5M
```

Click **Return**.

- l) Click **Apply** to change the tablespace size.
- 3) Go back to the **Automatic Undo Management** page to see the results of the changes you just made. You see that the undo retention time has increased to support the 48 hours requirement. Your undo tablespace size has also increased based on the changes you made to the size of the datafile for the undo tablespace.

Automatic Undo Management

In the General tab, you can view the current undo settings for your instance and use the Undo Advisor to analyze the undo tablespace requirements. This analysis can be performed based on the specified analysis period or the desired undo retention. The system activity for the specified time period can be viewed in the System Activity tab.

General **System Activity**

Undo Retention Settings	Undo Tablespace for this Instance
Undo Retention (minutes) 2880	Tablespace UNDOTBS1 Change Tablespace
Retention Guarantee No	Size (MB) 249
	Auto-Extensible Yes

- a) *Question:* Which Flashback operations are potentially affected by this change?
Answer: Flashback query, Flashback transaction, and Flashback table.
- b) *Question:* Do undo data survive the shutdown of a database?
Answer: Yes, undo is persistent.

Practices for Lesson 11

Background: You have just been informed of suspicious activities in the HR.JOBS table in your `orcl` database. The highest salaries seem to fluctuate in a strange way. You decide to enable standard database auditing and monitor data manipulation language (DML) activities in this table.

Practice 11-1: Configuring Database Auditing

Log in as the DBA1 user (with oracle_4U password, connect as SYSDBA) and perform the necessary tasks either through Enterprise Manager Database Control or through SQL*Plus. All scripts for this practice are in the /home/oracle/labs directory.

- 1) Use Enterprise Manager to enable database auditing. Set the **AUDIT_TRAIL** parameter to **XML**.
 - a) Invoke Enterprise Manager as the **DBA1** user in the **SYSDBA** role for your **orcl** database.
 - b) Click the **Server** tab, and then click **Audit Settings** in the Security section.
 - c) Click the value of Audit Trail, the **DB** link.
 - d) On the Initialization Parameters page, click the **SPFILE** tab.
 - e) Enter **audit** in the Name field and then click **Go**.
 - f) For the **audit_trail** parameter, enter **XML** as the value.
 - g) Click **Show SQL**.

```
Show SQL
ALTER SYSTEM SET audit_trail = "XML" SCOPE=SPFILE
Return
```

- h) Review the statement and then click **Return**.
 - i) On the Initialization Parameters page, click **Apply**.
- 2) Because you changed a static parameter, you must restart the database. Do so by running the **lab_11_01_02.sh** script.
 - a) In a terminal window, enter:

```
$ cd /home/oracle/labs
$ ./lab_11_01_02.sh
```
 - b) Continue with the next step when you see that the database is restarted.
- 3) Back in Enterprise Manager, select **HR.JOB\$** as the audited object and **DELETE**, **INSERT**, and **UPDATE** as Selected Statements. Gather audit information by session. Because the database has been restarted, you have to log in to Enterprise Manager again as the **DBA1** user.
 - a) Click **logout** in the upper-right corner of the Enterprise Manager window.
 - b) Log in as the **DBA1** user in the **SYSDBA** role for your **orcl** database.
 - c) Click the Database home page tab to ensure that Enterprise Manager had time to update the status of the database and its agent connections.
 - d) Click the **Server** tab, and then click **Audit Settings** in the Security section.
 - e) Click the **Audited Objects** tab at the bottom of the page, and then click the **Add** button.

Practice 11-1: Configuring Database Auditing (continued)

- f) On the Add Audited Object page, ensure that the Object Type is **Table**, and enter **HR.JOBS** in the Table field (or use the flashlight icon to retrieve this table).
- g) Move **DELETE**, **INSERT**, and **UPDATE** into the Selected Statements area by double-clicking each of them.
- h) Click **Show SQL**.

Show SQL	Return
<code>AUDIT DELETE, INSERT, UPDATE ON HR.JOBS BY SESSION</code>	

- i) Review the statement, and then click **Return**.
- j) Click **OK** to activate this audit.
- 4) Provide input for the audit, by executing the **lab_11_01_04.sh** script. This script creates the AUDIT_USER user, connects to SQL*Plus as this user, and multiplies the values in the MAX_SALARY column by 10. Then the HR user connects and divides the column values by 10. Finally, the AUDIT_USER user is dropped again.
- a) In a terminal window, enter:

```
$ cd /home/oracle/labs
$ ./lab_11_01_04.sh
```

- 5) In Enterprise Manager, review the audited objects.
 - a) Click the **Server** tab, and then click **Audit Settings** in the Security section.
 - b) Click **Audited Objects** in the Audit Trails area, which is on the right side of the page.
 - c) On the Audited Objects page, review the collected information.

Audited Objects	Filter Result	Return		
Show SQL				
Schema	Object Name	User Name	Action	Time (In Session's Time Zone)
HR	JOBS	HR	SESSION REC	2009-07-09 08:57:01.812159
HR	JOBS	AUDIT_USER	SESSION REC	2009-07-09 08:57:01.71946

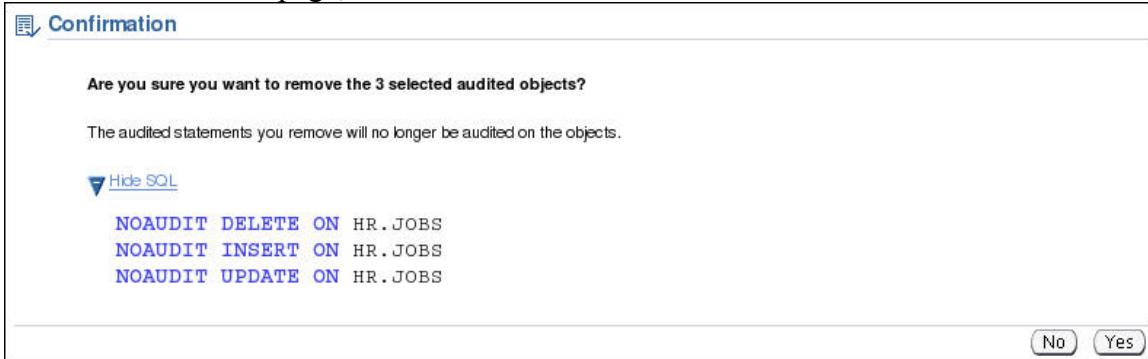
Question: Can you tell which user increased and which user decreased the salaries?

Answer: No, the standard audit records only show which user accessed the table.

- d) Click **Return**.
- 6) Undo your audit settings for HR.JOBS, disable database auditing, and then restart the database by using the **lab_11_01_06.sh** script.
 - a) On the Audit Settings page, click the **Audited Objects** tab at the bottom of the page.
 - b) Enter **HR** as Schema, and then click **Search**.
 - c) Select all three rows, and then click **Remove**.

Practice 11-1: Configuring Database Auditing (continued)

d) On the Confirmation page, click **Show SQL**.



The screenshot shows a confirmation dialog box with the following content:

Confirmation

Are you sure you want to remove the 3 selected audited objects?

The audited statements you remove will no longer be audited on the objects.

Hide SQL

NOAUDIT DELETE ON HR.JOBS
NOAUDIT INSERT ON HR.JOBS
NOAUDIT UPDATE ON HR.JOBS

No **Yes**

e) Review the statements, and then click **Yes** to confirm your removal.

f) On the Audit Settings page, click **XML** in the Configuration region.

g) On the Initialization Parameters page, click the **SPFile** tab.

h) On the SPFile page, enter **audit** in the Name field, and then click **Go**.

i) For the **audit_trail** parameter, enter **DB** as the value.

j) Click **Show SQL**.



The screenshot shows the SPFile page with the following content:

Show SQL

ALTER SYSTEM SET audit_trail = "DB" SCOPE=SPFILE

Return

k) Review the statement, and then click **Return**.

l) On the Initialization Parameters page, click **Apply**.

m) Because you changed a static parameter, you must restart the database. Do so by running the **lab_11_01_06.sh** script. In a terminal window, enter:

```
$ cd /home/oracle/labs  
$ ./lab_11_01_06.sh
```

7) Maintain your audit trail: Because you are completely finished with this task, backup and delete all audit files from the **/u01/app/oracle/admin/orcl/adump** directory.

a) In a terminal window, enter:

```
$ cd /u01/app/oracle/admin/orcl/adump  
$ ls
```

b) Create a backup of the audit trail files, and then remove the files

```
$ tar -czf $HOME/audit_today.tar.z *  
$ rm -f *
```

c) Close the terminal window.

Practices for Lesson 12

Background: You want to proactively monitor your `orcl` database so that common problems can be fixed before they affect users. This practice session invents some issues so that you can familiarize yourself with the tools that are available. First, execute scripts to set up your database environment for this exercise.

Practice 12-1: Database Maintenance

- 1) Create a new, locally managed tablespace called TBSSPC. It has a data file of 50 MB in the +DATA disk group. Ensure that the TBSSPC tablespace does not use Automatic Segment Space Management (ASSM). Execute the `lab_12_01_01.sh` script to perform these tasks. In a terminal window, enter:

```
$ cd /home/oracle/labs
$ cat lab_12_01_01.sh
...
sqlplus / as sysdba << END

set echo on

drop tablespace TBSSPC including contents and datafiles;

CREATE SMALLFILE TABLESPACE "TBSSPC"
DATAFILE '+DATA' SIZE 50M
AUTOEXTEND ON NEXT 10M MAXSIZE 200M
LOGGING
EXTENT MANAGEMENT LOCAL
SEGMENT SPACE MANAGEMENT MANUAL;

exit;
END

$ ./lab_12_01_01.sh
```

- 2) Create a new SPCT user, identified by `oracle_4U`. Assign the TBSSPC tablespace as the default tablespace. Assign the TEMP tablespace as the temporary tablespace. Grant the following roles to the SPCT users: CONNECT, RESOURCE, and DBA. Execute the `lab_12_01_02.sh` script to perform these tasks. In a terminal window, enter:

```
$ cat lab_12_01_02.sh
...
sqlplus / as sysdba << END

set echo on

drop user spct cascade;

create user spct identified by oracle_4U
default tablespace TBSSPC
temporary tablespace temp;

grant connect, resource, dba to spct;

exit;
END

$ ./lab_12_01_02.sh
```

Practice 12-1: Database Maintenance (continued)

3) Use the DBMS_ADVVISOR package to set the database activity time to 30 minutes. As the SPCT user, drop and create the SPCT table and gather statistics for this table. Create a snapshot in Automatic Workload Repository (AWR). Execute the **lab_12_01_03.sh** script to perform these tasks. In a terminal window, enter:

```
$ cat lab_12_01_03.sh
...
sqlplus / as sysdba << EOF

set echo on

exec
dbms_advisor.set_default_task_parameter('ADDM', 'DB_ACTIVITY_MI
N', 30);

connect spct/oracle_4U

drop table spct purge;
create table spct(id number, name varchar2(2000));

exec DBMS_STATS.GATHER_TABLE_STATS(-
ownname=>'SPCT', tabname=>'SPCT', -
estimate_percent=>DBMS_STATS.AUTO_SAMPLE_SIZE);

exec DBMS_WORKLOAD_REPOSITORY.CREATE_SNAPSHOT();

exit;
EOF
$ ./lab_12_01_03.sh
```

4) Create the activity to be analyzed. Execute the **lab_12_01_04.sh** script to perform these tasks.
In a terminal window, enter the following. You may have to press [Enter] after you see that several PL/SQL procedures have completed, in order to see the command prompt again.

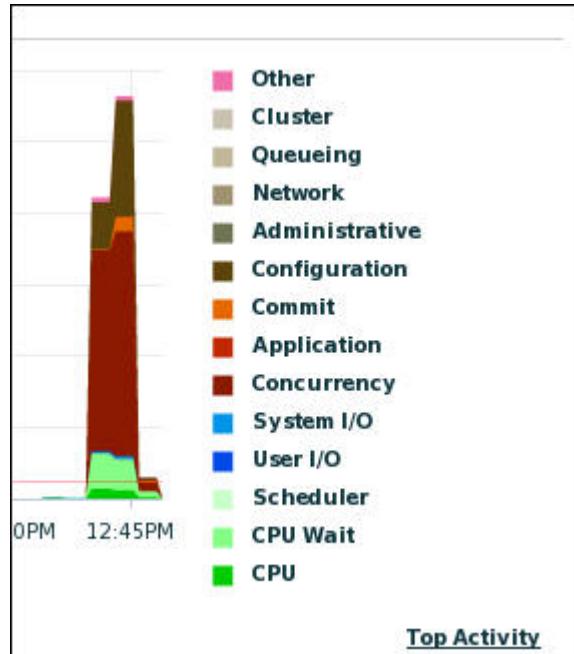
```
$ ./lab_12_01_04.sh
```

5) In Enterprise Manager, review the Performance page as a user connected as SYSDBA. View performance data in real time with a 15-seconds refresh cycle. After a while, you should see a spike on the Average Active Sessions graph. This is your activity to be analyzed. Looking at the graph, you can already determine that this instance is suffering from concurrency problems.

- Invoke Enterprise Manager as the **DBA1** user in the **SYSDBA** role for your **orcl** database.

Practice 12-1: Database Maintenance (continued)

b) Click the **Performance** tab.



Note: Depending on when you run the workload, you may see differences between your graph and the one provided as a possible solution.

c) After the spike is finished, execute the **lab_12_01_05.sh** script. This script forces the creation of a new snapshot and gathers statistics on your SPCT table.

Note: Causing the same performance problem in all environments is not easy. To help make your test more successful, wait an extra minute or so after the spike has completely finished before running the script.

After the spike has finished, in a terminal window, enter:

```
$ ./lab_12_01_05.sh
```

6) Look at the **Performance Analysis** findings in order of their impact. There are several access paths to this information. The results should look similar to the following:

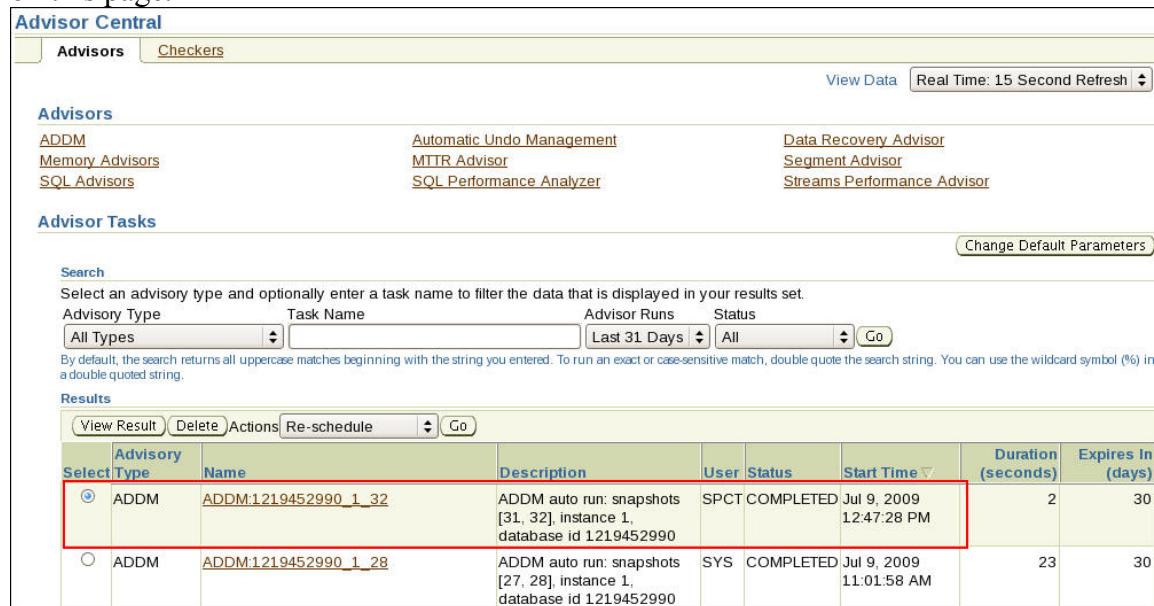
ADDM Performance Analysis					
Task Name	SPCT	Average Active Sessions	Period Start Time	Period	Duration
ADDM:1219452990_1_32		8.1	Jul 9, 2009 12:42:25 PM GMT+07:00	5 (minutes)	
Filters View Snapshots View Report					
Impact (%) ▾	Finding	Occurrences (24 hrs ending with analysis period)			
95.9	Top SQL Statements	4 of 22			
32.1	Buffer Busy - Hot Objects	1 of 22			
21.6	Unusual "Concurrency" Wait Event	1 of 22			
19.8	High Watermark Waits	1 of 22			
10.5	Buffer Busy - Hot Block	1 of 22			
4.2	Shared Pool Latches	1 of 22			
2.4	Commits and Rollbacks	2 of 22			
2	Buffer Cache Latches	1 of 22			

Practice 12-1: Database Maintenance (continued)

Looking at the Performance Analysis section, you see that the first finding has a high percentage (in this example, 95.9%) impact on the system. So your first impulse is to look at this finding in more detail. However, looking at this SQL statement does not yet help you to understand the concurrency problem of your database.

Investigate the other findings in order of severity. Look at the Buffer Busy findings in particular. For one of the Buffer Busy results, you should see that there is read-and-write contention on your SPCT table. The recommended action is to use the Automatic Segment Space Management (ASSM) feature for your SPCT table. The following steps guide you through this exercise.

- a) Navigate to the Database home page, and then click **Advisor Central** at the bottom of the page.
- b) Your ADDM task should already be displayed. If not, search for it and display it on this page.



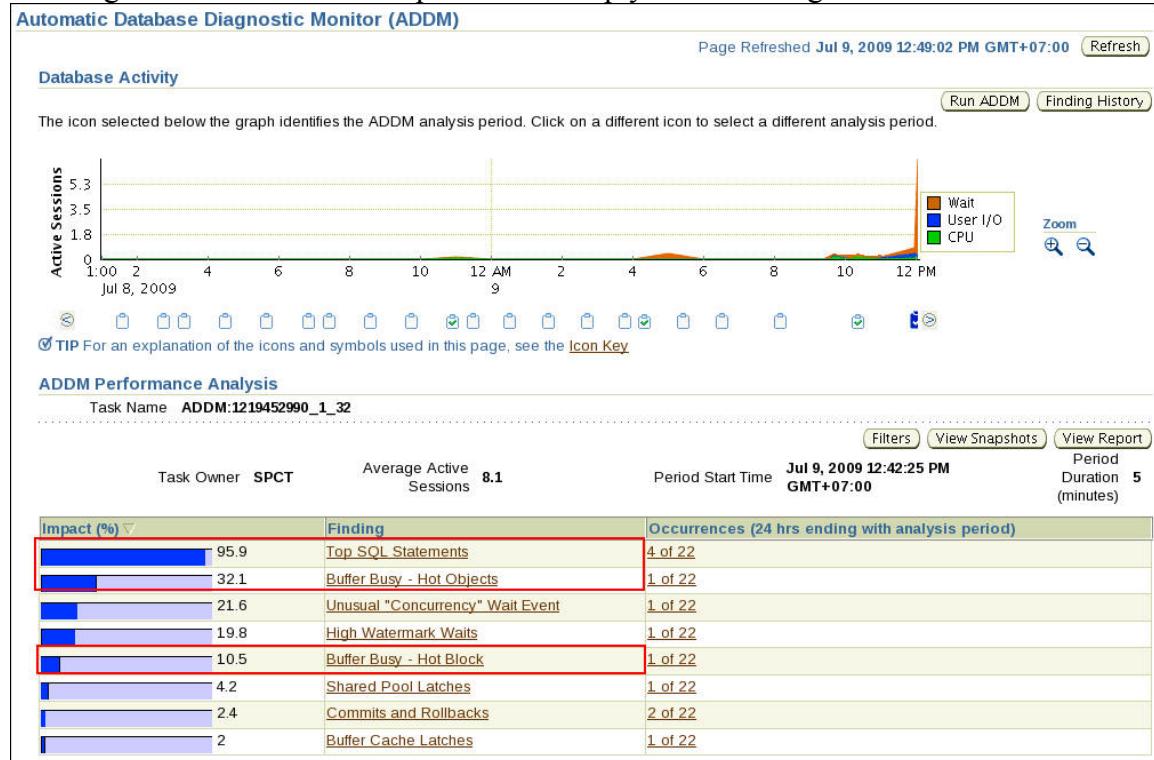
The screenshot shows the Advisor Central interface. The 'Advisors' tab is selected. Under 'Advisors', there are links for ADDM, Memory Advisors, and SQL Advisors. Under 'Automatic Undo Management', there are links for MTTR Advisor and SQL Performance Analyzer. On the right, there are links for Data Recovery Advisor, Segment Advisor, and Streams Performance Advisor. The 'View Data' button and 'Real Time: 15 Second Refresh' button are at the top right. The 'Advisor Tasks' section has a 'Search' field and filters for 'Advisory Type' (All Types), 'Task Name' (Last 31 Days), 'Advisor Runs' (All), and 'Status' (Go). Below this, a note says: 'By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.' The 'Results' section has a table with columns: Select, Advisory Type, Name, Description, User, Status, Start Time, Duration (seconds), and Expires In (days). Two rows are shown: one for ADDM task ADDM:1219452990_1_32 (Status: COMPLETED, Duration: 2 seconds, Expires In: 30 days) and one for ADDM task ADDM:1219452990_1_28 (Status: COMPLETED, Duration: 23 seconds, Expires In: 30 days). The first row is highlighted with a red box.

Select	Advisory Type	Name	Description	User	Status	Start Time	Duration (seconds)	Expires In (days)
<input checked="" type="radio"/>	ADDM	ADDM:1219452990_1_32	ADDM auto run: snapshots [31, 32], instance 1, database id 1219452990	SPCT	COMPLETED	Jul 9, 2009 12:47:28 PM	2	30
<input type="radio"/>	ADDM	ADDM:1219452990_1_28	ADDM auto run: snapshots [27, 28], instance 1, database id 1219452990	SYS	COMPLETED	Jul 9, 2009 11:01:58 AM	23	30

- c) Select the task, and then click the **View Result** button (alternatively, click the name of the task).

Practice 12-1: Database Maintenance (continued)

d) The ADDM page appears, showing the detailed results from the ADDM run.
Note: Do not click the Run ADDM Now button because you already executed the ADDM performance analysis when you ran the `lab_12_01_05.sh` script. Clicking the button now would produce an empty set of findings.



Looking at the Performance Analysis section, you see that the first finding has a high impact on the system (in this example, 95.9%). You also notice that there are Buffer Busy findings as well. Because the Top SQL Statements finding is impacting your system by such a high percent, your first impulse is to look at this finding in more detail.

Note: Because there are several factors that can affect performance on your classroom machine, your results may not be exactly as shown. The findings may appear in a different order. If you do not see results similar to the ones outlined in the preceding screenshot, you may need to restart this practice. If you still do not see the expected results, you may need to adjust the load by modifying the `lab_12_01_04.sh` and `lab_12_01_04.sql` scripts. Ask your instructor for assistance if this is the case. Take care not to increase the load too much or you will slow your system down too much.

- Click the “Top SQL Statements” link in the Finding column.
- Review the recommendations on the Performance Finding Details page. However, looking at this SQL statement does not yet help you to understand the concurrency problem of your database. Click the **Back** icon in your Web browser.

Practice 12-1: Database Maintenance (continued)

g) Look at the Buffer Busy findings on the Automatic Database Diagnostic Monitor (ADDM) page. Click the first occurrence of the Buffer Busy finding, in this case, the Buffer Busy – Hot Objects link.

Performance Finding Details: Buffer Busy - Hot Objects

Finding **Read and write contention on database blocks was consuming significant database time.** ([Finding History](#))

Impact (Active Sessions) **2.59**  32.1

Percentage of Finding's Impact (%) **32.1** 

Period Start Time **Jul 9, 2009 12:42:25 PM GMT+07:00**

Period Duration (minutes) **5**

Filtered **No** ([Filters](#))

Recommendations

Show All Details | Hide All Details

Details	Category	Benefit (%)
▼ Hide	Schema	32.1

Action Consider using ORACLE's recommended solution of automatic segment space management in a locally managed tablespace for the tablespace "TBSSPC" containing the TABLE "SPCT.SPCT" with object ID 74614. Alternatively, you can move this object to a different tablespace that is locally managed with automatic segment space management.

Database Object [SPCT.SPCT](#)

Rationale There was significant read and write contention on TABLE "SPCT.SPCT" with object ID 74614.

Database Object	Benefit (%)
SPCT.SPCT	32.1
Show Schema	32.1
Show Schema	32.1

Findings Path

Expand All | Collapse All

Findings	Percentage of Finding's Impact (%)	Additional Information
▼ Read and write contention on database blocks was consuming significant database time.	32.1	
▼ Read and write contention on database blocks was consuming significant database time.	32.1	
Wait class "Concurrency" was consuming significant database time.	59.8	

The findings show that there is read-and-write contention on database blocks. The recommendation is to use a tablespace that is locally managed with Automatic Segment Space Management.

h) Go back to the ADDM page and look at the other Buffer Busy findings. One of them should look similar to the following:

Performance Finding Details: Buffer Busy - Hot Block

Finding **A hot data block with concurrent read and write activity was found. The block belongs to segment "SPCT.SPCT" and is block 631 in file 8.** ([Finding History](#))

Impact (Active Sessions) **.85**  10.5

Percentage of Finding's Impact (%) **10.5** 

Period Start Time **Jul 9, 2009 12:42:25 PM GMT+07:00**

Period Duration (minutes) **5**

Filtered **No** ([Filters](#))

Recommendations

Show All Details | Hide All Details

Details	Category	Benefit (%)
▼ Hide	Application Analysis	10.5

Action Investigate application logic to find the cause of high concurrent read and write activity to the data present in this block.

Rationale The SQL statement with SQL_ID "3csh3g3mjhzmz" spent significant time on "buffer busy" waits for the hot block.

SQL Text	SQL ID
INSERT INTO SPCT VALUES (NULL 'a')	3csh3g3mjhzmz

Show Schema	Benefit (%)
Show Schema	10.5

Findings Path

Expand All | Collapse All

Findings	Percentage of Finding's Impact (%)	Additional Information
▼ A hot data block with concurrent read and write activity was found. The block belongs to segment "SPCT.SPCT" and is block 631 in file 8.	10.5	
▼ Read and write contention on database blocks was consuming significant database time.	32.1	
Wait class "Concurrency" was consuming significant database time.	59.8	

This finding shows that there is a hot data block that belongs to the SPCT . SPCT table. The recommendation is to investigate the application logic to find the cause.

Practice 12-1: Database Maintenance (continued)

7) You decide to implement the recommendation to use Automated Segment Space Management. To do this, you must re-create the object. Create a new, locally managed tablespace, called TBSSPC2 with a 50 MB data file in the +DATA disk group. Ensure that the TBSSPC2 tablespace uses the Automatic Segment Space Management feature. Then execute the `lab_12_01_07.sh` script to drop the SPCT table, to re-create it in the new tablespace, to gather statistics, and to take a new snapshot.

- In Enterprise Manager, click the **Server** tab, and then **Tablespaces** in the Storage section.
- Click **Create**.
- Enter TBSSPC2 as the tablespace name, and verify that Extent Management is **Locally Managed**, Type is **Permanent**, Status is **Read Write**, and Use bigfile tablespace is *not* selected.
- Click **Add** in the **Datafiles** region.
- On the Add Datafile page, ensure that the DiskGroup is **DATA** and enter **50 MB** as File Size.
- Click **Continue**.
- Click the **Storage** tab, and verify that Extent Allocation is **Automatic**, Segment Space Management is **Automatic**, and Logging is **enabled**.
- Click the **General** tab.
- Click **Show SQL**, and view the SQL that will be run, and then click **Return**.

```
Show SQL
Return
CREATE SMALLFILE TABLESPACE "TBSSPC2" DATAFILE '+DATA' SIZE 50M LOGGING EXTENT MANAGEMENT LOCAL SEGMENT
SPACE MANAGEMENT AUTO
```

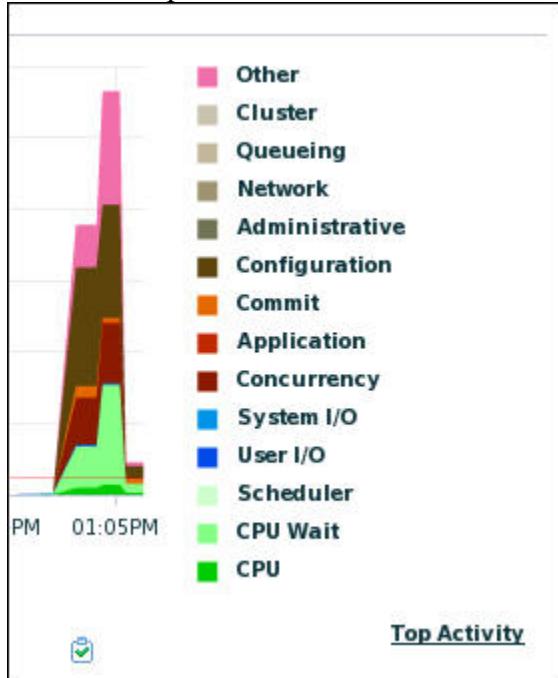
- Click **OK**. A successful Confirmation message appears.
- In a terminal window, enter:

```
$ ./lab_12_01_07.sh
```
- Execute your workload again. (Use the `lab_12_01_04.sh` script.)
In a terminal window, enter the following. You may have to press [Enter] after you see that several PL/SQL procedures have completed, in order to see the command prompt again.

```
$ ./lab_12_01_04.sh
```
- In Enterprise Manager, review the Performance page as a user connected as SYSDBA. View performance data in real time with a 15-seconds refresh cycle. After a while, you should see a spike on the Average Active Sessions graph.
After the spike is finished, execute the `lab_12_01_05.sh` script again. This script forces the creation of a new snapshot and gathers statistics on your ADDM table.

Practice 12-1: Database Maintenance (continued)

- a) Invoke Enterprise Manager as the **DBA1** user in the **SYSDBA** role for your **orcl** database.
- b) Click the **Performance** tabbed page. Watch for the spike in the Active Sessions chart to complete.



- c) After the spike is finished, run the **lab_12_01_05.sh** script to force the creation of a new snapshot and gather statistics on your SPCT table. Enter the following in a terminal window:

```
$ ./lab_12_01_05.sh
```

- 10) Review the ADDM from the **Advisor Central** link.

- a) Navigate to the Database home page, and then click **Advisor Central** at the bottom of the page.
- b) Click the top-most ADDM task name.

Practice 12-1: Database Maintenance (continued)

c) You see that the Buffer Busy findings about the read-and-write contention is no longer there. By moving the ADDM table to the locally managed TBSSPC2 tablespace, which uses the Automatic Autoextend Segment feature, you obviously fixed the root cause of the contention problem.

Note: You may see additional Buffer Busy findings (at a lower impact percentage) and other further recommendations that could improve performance, but you are not going to pursue them at this time.



11) To not affect other practice sessions, execute the **lab_12_01_11.sh** script to clean up your environment.

In a terminal window, enter:

```
$ ./lab_12_01_11.sh
```

Practices for Lesson 13

Background: Users are complaining about slower-than-normal performance for operations involving the human resources and order-entry applications. When you question other members of the DBA staff, you find that maintenance was recently performed on some of the tables belonging to the HR schema. You need to troubleshoot and make changes as appropriate to resolve the performance problems. SQL script files are provided for you in the /home/oracle/labs directory. Other directories are individually named.

Practice 13-1: Managing Performance

- 1) Log in to SQL*Plus as the SYS user and perform maintenance on tables in the HR schema by running the **lab_13_01_01.sql** script.

```
$ cd ~/labs
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ sqlplus / as sysdba
SQL> @lab_13_01_01.sql
```

- 2) You get calls from HR application users saying that a particular query is taking longer than normal to execute. The query is in the **lab_13_01_02.sql** script. To run this file, enter the following in SQL*Plus:

```
SQL> CONNECT hr
Password: oracle_4U    <<<Password does not appear on screen
Connected.
SQL> @lab_13_01_02.sql
```

- 3) Using Enterprise Manager, locate the HR session in which the above statement was just executed, and view the execution plan for that statement.
 - a) In Enterprise Manager, click the **Performance** tab, and the click **Search Sessions** in the Additional Monitoring Links section.
 - b) On the Search Sessions page, change the Search criteria to “**DB User**,” enter **HR** in the field to the right of that, and then click **Go**.
 - c) Click the **SID** number in the Results listing.
 - d) You now see the Session Details page for this session. Click the **hash value link** to the right of the Previous SQL label in the Application section.

Practice 13-1: Managing Performance (continued)

e) On the SQL Details page, you see the details for the last SQL statement executed by that session, which is the one in question. Click the **Plan** tab to see the execution plan for the query.

The screenshot shows the 'SQL Details' page for session 'bckcqw5pd108'. The 'Plan' tab is selected. The query is:

```
select *  
from hr.employees  
where employee_id = 200
```

The execution plan table is as follows:

Operation	Object	Order	Rows	Bytes	Cost	CPU (%)	Time	Query Block Name/Object Alias	Filter	Projection
SELECT STATEMENT		2			3	100				
TABLE ACCESS FULL	EMPLOYEES	1	1	69	3	0	0:0:1	SEL\$1 / EMPLOYEES@SEL\$1	"EMPLOYEE_ID"=200	"EMPLOYEE_ID"=200 [NUMBER,22], "EMP..."

[Show Explain Rewrite](#)

f) You see in the Operation column that this query is doing a full table scan (TABLE ACCESS FULL). Because you know that the query's condition is an equality comparison on the primary key (EMPLOYEE_ID), you decide to investigate the status of the primary key index.

4) Using Enterprise Manager, check to see the status of the **EMPLOYEE** table's index on **EMPLOYEE_ID**. See if it is **VALID**.

- From the Database Home page, click the **Schema** tab, and then **Indexes**.
- Select **Table Name** as the Search By value.
- Enter **HR** in the **Schema** field.
- Enter **EMPLOYEES** for Object Name.
- Click **Go**, and the list of six indexes appears.

Practice 13-1: Managing Performance (continued)

f) Click the index named **EMP_EMP_ID_PK**.

Select	Table Owner	Table	Indexed Columns	Index Owner	Index	Table Type	Tablespace	Partitioned	Last Analyzed
<input checked="" type="checkbox"/>	HR	EMPLOYEES	EMAIL	HR	EMP_EMAIL_UK	TABLE	EXAMPLE	NO	Aug 26, 2008 12:46:38 AM GMT+07:00
<input checked="" type="checkbox"/>	HR	EMPLOYEES	MANAGER_ID, LAST_NAME	HR	EMP_NAME_IX	TABLE	EXAMPLE	NO	Aug 26, 2008 12:46:38 AM GMT+07:00
<input checked="" type="checkbox"/>	HR	EMPLOYEES	MANAGER_ID	HR	EMP_MANAGER_IX	TABLE	EXAMPLE	NO	Aug 26, 2008 12:46:38 AM GMT+07:00
<input checked="" type="checkbox"/>	HR	EMPLOYEES	DEPARTMENT_ID	HR	EMP_DEPARTMENT_IX	TABLE	EXAMPLE	NO	Aug 26, 2008 12:46:38 AM GMT+07:00
<input checked="" type="checkbox"/>	HR	EMPLOYEES	JOB_ID	HR	EMP_JOB_IX	TABLE	EXAMPLE	NO	Aug 26, 2008 12:46:38 AM GMT+07:00
<input checked="" type="checkbox"/>	HR	EMPLOYEES	EMPLOYEE_ID	HR	EMP_EMP_ID_PK	TABLE	EXAMPLE	NO	Aug 26, 2008 12:46:38 AM GMT+07:00

g) On the View Index page, notice that the status of the index is UNUSABLE.

General	
Name	EMP_EMP_ID_PK
Schema	HR
Tablespace	EXAMPLE
Index Type	Normal
Status	UNUSABLE

5) Now that you have seen one index with a non-VALID status, you decide to check all indexes. Using SQL*Plus, as the **HR** user, find out which HR schema indexes do not have STATUS of VALID. To do this, you can query a data dictionary view with a condition on the STATUS column.

a) Go to the SQL*Plus session where you are still logged in as the HR user, and run this query:

```
SQL> select index_name, table_name, status
  from user_indexes where status <> 'VALID';
```

INDEX_NAME	TABLE_NAME	STATUS
EMP_EMAIL_UK	EMPLOYEES	UNUSABLE
EMP_EMP_ID_PK	EMPLOYEES	UNUSABLE
EMP_DEPARTMENT_IX	EMPLOYEES	UNUSABLE

Practice 13-1: Managing Performance (continued)

EMP_JOB_IX	EMPLOYEES	UNUSABLE
EMP_MANAGER_IX	EMPLOYEES	UNUSABLE
EMP_NAME_IX	EMPLOYEES	UNUSABLE

6 rows selected.

SQL>

- b) You notice that the output lists six indexes, all on the **EMPLOYEES** table. This is a problem you will need to fix.
- 6) You decide to use Enterprise Manager to reorganize all the indexes in the **HR** schema that are marked as **UNUSABLE**.
 - a) In Enterprise Manager, on the page displaying the **EMP_EMP_ID_PK** index, select **Reorganize** in the Actions list, and then click **Go**.
 - b) On the Reorganize Objects pages, click **Add**, to add each of the other five indexes to the reorganization operation.
 - c) In the Add screen, choose **Indexes** for the Type drop-down list, and enter **HR** in the Schema field. Click **Search**.
 - d) Select the five other indexes whose names start with “**EMP_**.”

Objects: Add

Database Instance **orcl.oracle.com** Logged In As **SYS** Cancel OK

Search

Type	Indexes	<input checked="" type="checkbox"/> Include minimum size filter in search query
Schema	HR	Minimum Size (KB)
Object Name		
Partition Name		
Tablespace		
Search		

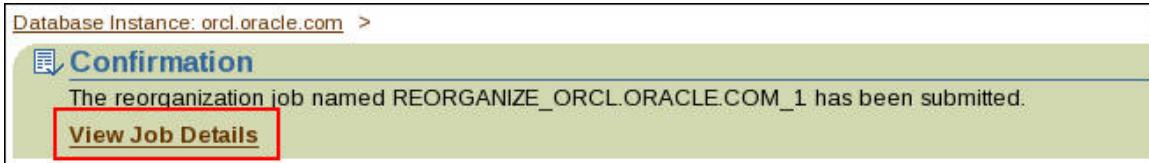
Available Objects: Indexes

Select	Name	Type	Tablespace	Size (KB)	Partitioned	IOT
<input type="checkbox"/>	HR.DEPT_ID_PK	Index	EXAMPLE	64	NO	N/A
<input type="checkbox"/>	HR.DEPT_LOCATION_IX	Index	EXAMPLE	64	NO	N/A
<input checked="" type="checkbox"/>	HR.EMP_DEPARTMENT_IX	Index	EXAMPLE	64	NO	N/A
<input checked="" type="checkbox"/>	HR.EMP_EMAIL_UK	Index	EXAMPLE	64	NO	N/A
<input checked="" type="checkbox"/>	HR.EMP_JOB_IX	Index	EXAMPLE	64	NO	N/A
<input checked="" type="checkbox"/>	HR.EMP_MANAGER_IX	Index	EXAMPLE	64	NO	N/A
<input checked="" type="checkbox"/>	HR.EMP_NAME_IX	Index	EXAMPLE	64	NO	N/A
<input type="checkbox"/>	HR.JHIST_DEPARTMENT_IX	Index	EXAMPLE	64	NO	N/A
<input type="checkbox"/>	HR.JHIST_EMPLOYEE_IX	Index	EXAMPLE	64	NO	N/A
<input type="checkbox"/>	HR.JHIST_EMP_ID_ST_DATE_PK	Index	EXAMPLE	64	NO	N/A

- e) Click **OK** to go back to the Reorganize Objects: Objects page.
- f) Verify that all six indexes for the **EMPLOYEES** table are listed and click **Next**.
- g) Keep all the default settings for Options, and then click **Next**. The reorganize script generation occurs, and then the Impact Report appears.
- h) Note that there are no problems reported on Impact Report, and then click **Next**.

Practice 13-1: Managing Performance (continued)

- i) On the Schedule page, enter **oracle** and **oracle** for Username and Password under Host Credentials.
- j) Click **Next**.
- k) On the Review page, click **Submit Job**.
- l) After the Confirmation page appears, click the **View job Details** to see the job status.



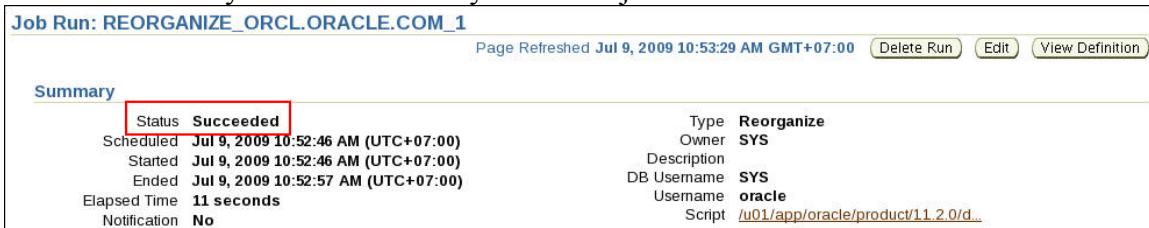
Database Instance: orcl.oracle.com >

Confirmation

The reorganization job named REORGANIZE_ORCL.ORACLE.COM_1 has been submitted.

View Job Details

- m) Click **Reload** on your browser until you see the job has succeeded.



Job Run: REORGANIZE_ORCL.ORACLE.COM_1

Page Refreshed Jul 9, 2009 10:53:29 AM GMT+07:00 [Delete Run](#) [Edit](#) [View Definition](#)

Summary

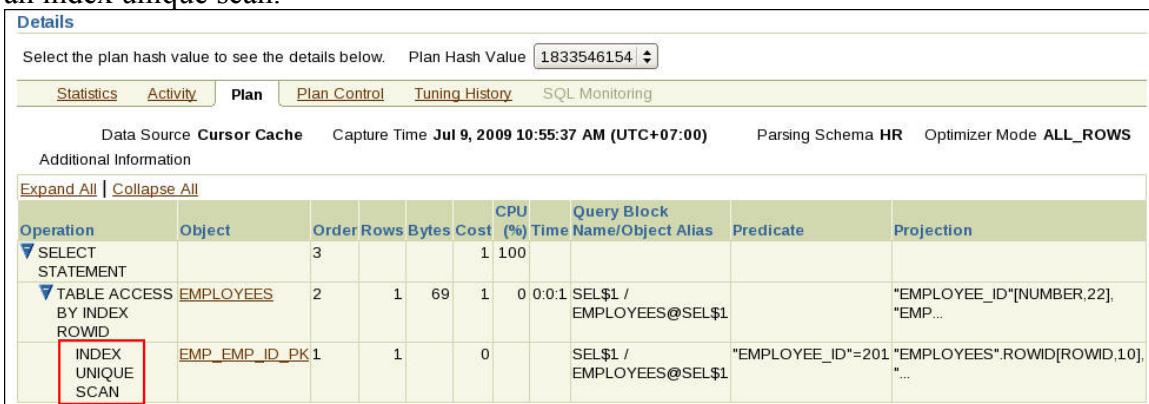
Status	Succeeded
Scheduled	Jul 9, 2009 10:52:46 AM (UTC+07:00)
Started	Jul 9, 2009 10:52:46 AM (UTC+07:00)
Ended	Jul 9, 2009 10:52:57 AM (UTC+07:00)
Elapsed Time	11 seconds
Notification	No
Type	Reorganize
Owner	SYS
Description	
DB Username	SYS
Username	oracle
Script	/u01/app/oracle/product/11.2.0/d...

- 7) Return to the SQL*Plus session where the HR user is logged in, and run the **lab_13_01_07.sql** script to execute the same kind of query. Then repeat the steps to see the plan of the last SQL statement executed by this session, to see if the plan has changed.

- a) Enter the following at the SQL*Plus prompt:

```
SQL> @lab_13_01_07.sql
```

- b) Repeat the tasks listed in step 3 to view the execution plan for the query. Now the icon indicates the use of an index. Click **View Table**. Note that the plan now uses an index unique scan.



Details

Select the plan hash value to see the details below. Plan Hash Value 1833546154

[Statistics](#) [Activity](#) [Plan](#) [Plan Control](#) [Tuning History](#) [SQL Monitoring](#)

Data Source **CURSOR Cache** Capture Time Jul 9, 2009 10:55:37 AM (UTC+07:00) Parsing Schema **HR** Optimizer Mode **ALL_ROWS**

Additional Information

[Expand All](#) [Collapse All](#)

Operation	Object	Order	Rows	Bytes	Cost	CPU (%)	Time	Query Block Name/Object Alias	Predicate	Projection
SELECT STATEMENT		3			1	100				
TABLE ACCESS BY INDEX ROWID	EMPLOYEES	2	1	69	1	0	0:0:1	SEL\$1 / EMPLOYEES@SEL\$1		"EMPLOYEE_ID"[NUMBER,22], "EMP...
INDEX UNIQUE SCAN	EMP_EMP_ID_PK1		1		0			SEL\$1 / EMPLOYEES@SEL\$1	"EMPLOYEE_ID"=201	"EMPLOYEES".ROWID[ROWID,10], "

- c) Quit the SQL*Plus session.
- 8) What is the difference in execution plans, and why?

Answer: The statement execution uses a unique index scan instead of a full table scan, because the index is usable after your index reorganization.

Practice 13-1: Managing Performance (continued)

9) Simulate a working load on your instance by running the `lab_13_01_09.sql` script as the SYS user. Please note the SID value that is reported.

SID value reported: _____

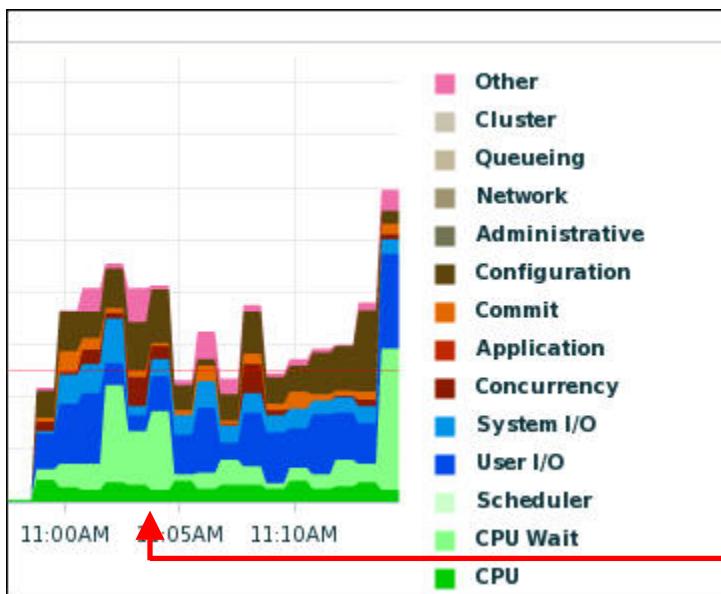
This script takes about 20 minutes to complete. So, run it in a separate terminal window and continue with this practice exercise while it runs.

Note: Because this script generates a fairly heavy load in terms of CPU and disk I/O, you will notice that response time for Database Control is slower.

```
$ sqlplus / as sysdba
SQL> @lab_13_01_09.sql
```

10) Go back to Enterprise Manager and examine the performance of your database.

- In Enterprise Manager, navigate to the Performance page, and investigate system performance.
- You may need to wait a minute or two to see the effects of the load generation script appear on the graphs.



Wait to see some levels of activity in the **Average Active Sessions** graph before proceeding.

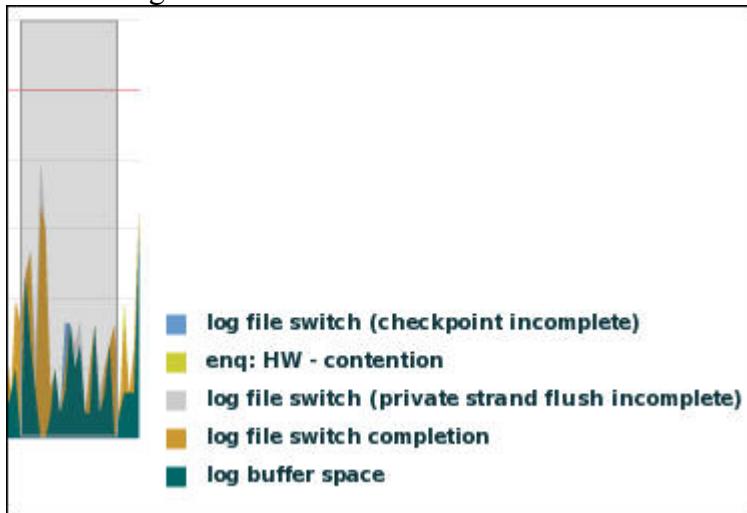
Question 1: In the Average Active Sessions graph, which are the two main categories that active sessions are waiting for?

Answer: In this example, it looks like CPU Wait and User I/O are quite high. Configuration is also showing high wait activity. Your results may differ from what is shown here.

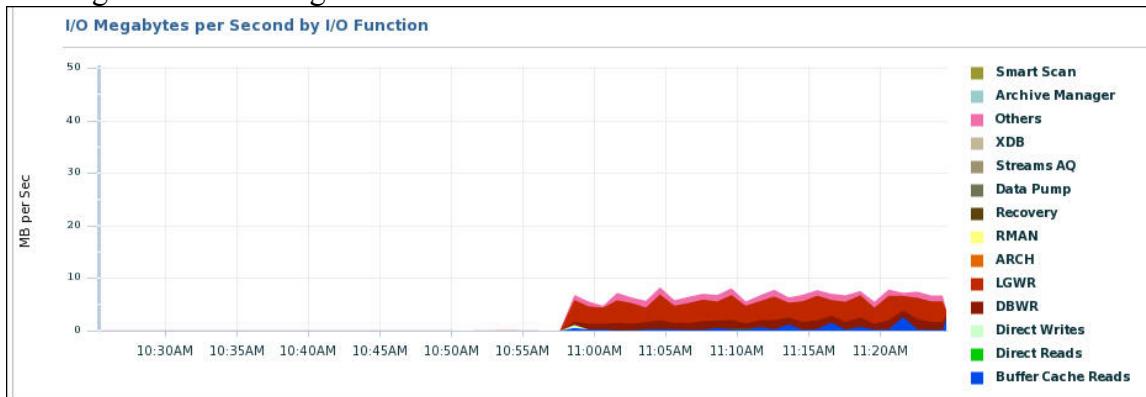
Question 2: In the Configuration category of waits, what is one of the contributors to the wait time? Click Configuration to see the graph.

Practice 13-1: Managing Performance (continued)

Answer: Any one of these, but log file switch completion and log buffer space seem to be the highest contributors:



Question 3: Click Back, and then click Settings on the Performance page. For the Detail Chart Settings select I/O for Default View, and I/O Function for I/O Chart Settings, and then click **OK**. Scroll down to the IO charts to determine which process is doing the most writing to the disk.

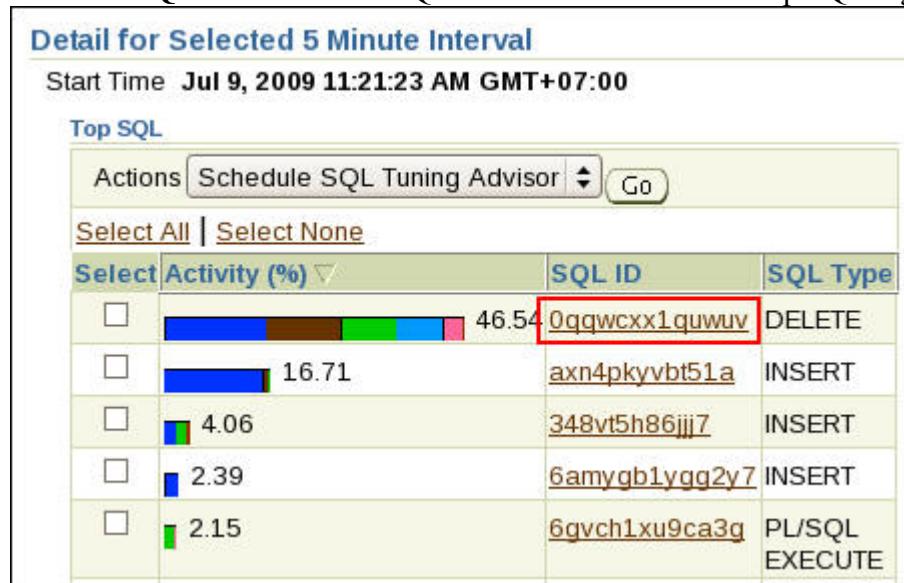


Answer: LGWR

c) Click **Top Activity** in the Additional Monitoring Links region.

Practice 13-1: Managing Performance (continued)

d) Click the SQL ID of the first SQL statement listed in the Top SQL region.



e) See the first SQL statement.

11) Kill the session that is generating the load. Use the session ID recorded in step 9. The session ID is listed in the **SID** column of the Detail for Selected 5 Minute Interval.

a) Click the **SID** number for the session ID recorded earlier. This is found under the heading **Detail for Selected 5 Minute Interval**.

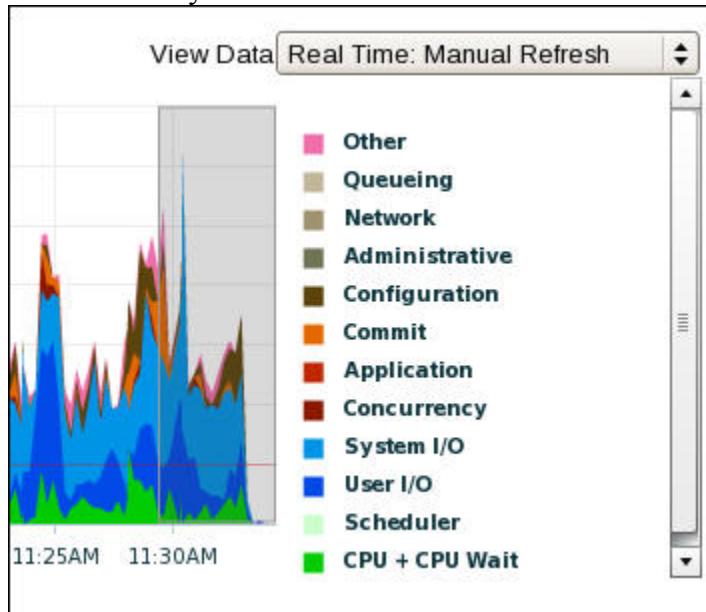


b) On the Session Details page, click **Kill Session**, and then click **Yes** to confirm.

Note: If you remain on this Session Details page long enough for a few automatic refreshes to be done, you may see a warning, "WARNING, Session has expired." or a SQL Error saying the session is marked for kill. This warning means you are attempting to refresh information about a session that has already been killed. You can ignore this warning.

Practice 13-1: Managing Performance (continued)

c) Click **Top Activity** in the navigation history at the top of the page. Note that the session activity in the database has declined considerably.



Practices for Lesson 14

Background: Your `orcl` database is ready to move from test or development into production.

Configure your database to reduce the chances of failure or data loss. To do so, perform the following tasks:

- Ensure redundancy of control files and backup the control file to trace
- Review the fast recovery area configuration
- Ensure that there are at least two redo log members in each group
- Place your database in `ARCHIVELOG` mode
- Configure redundant archive log destinations

Practice 14-1: Configuring Your Database for Recovery

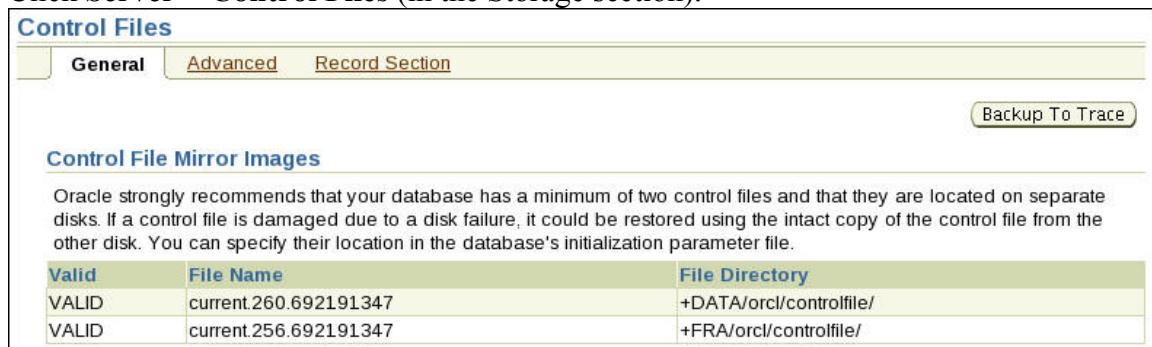
In this practice, you configure your database to reduce the chances of failure or data loss.

Note: Completing this practice is a prerequisite for all following backup and recovery practices.

- 1) First, run the **lab_14_01_01.sh** script to create some more data that will be used in scenarios during the upcoming practices. This script creates tables in the INVENTORY tablespace and simulates some basic activity on your database.

```
$ cd ~/labs
$ ./lab_14_01_01.sh
```

- 2) Verify that you have at least two control files to ensure redundancy.
 - a) Invoke Enterprise Manager as the **DBA1** user in the **SYSDBA** role for your **orcl** database.
 - b) Click **Server > Control Files** (in the Storage section).



The screenshot shows the 'Control Files' page in Oracle Enterprise Manager. The 'General' tab is selected. A table lists two control files:

Valid	File Name	File Directory
VALID	current.260.692191347	+DATA/orcl/controlfile/
VALID	current.256.692191347	+FRA/orcl/controlfile/

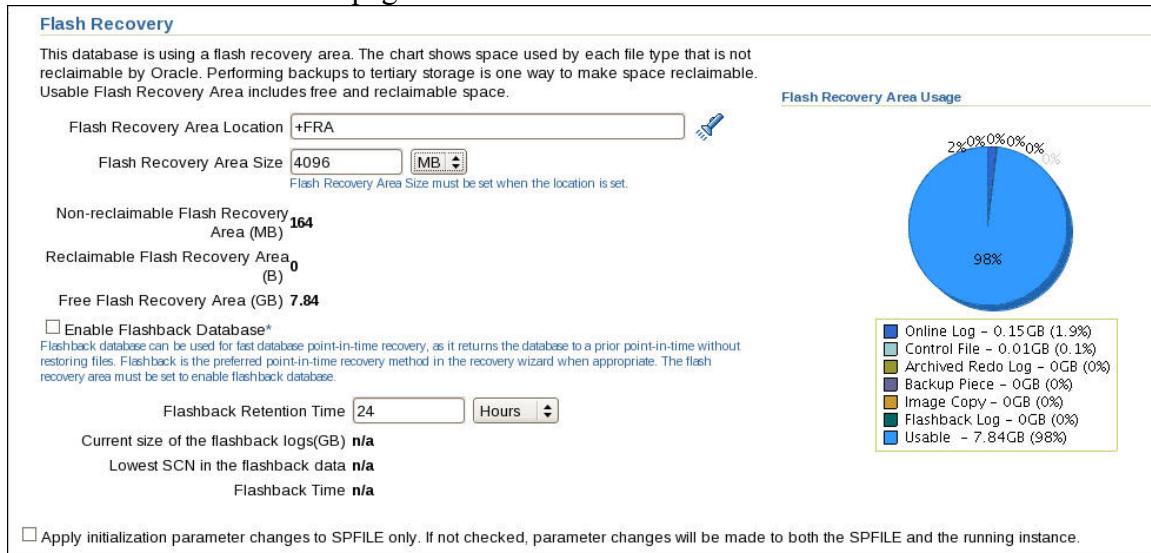
Question 1: On the Control Files: General page, how many control files do you have?

Answer: 2 .

- 3) Review the fast recovery area configuration and change the size to 8 GB.
 - a) In Enterprise Manager, select **Availability > Recovery Settings** in the Setup section.

Practice 14-1: Configuring Your Database for Recovery (continued)

b) Scroll to the bottom of the page.



c) **Question:** Is the fast recovery area enabled?

Answer: Yes, because the FRA was configured during database creation using dbca.

d) Note the location of the fast recovery area.

For example: +FRA

e) **Question:** Which essential DBA tasks can you perform in this section?

Answer: You can change the location, size or retention time for the fast recovery area, as well as enable the Flashback Database functionality.

f) **Question:** Does changing the size of the fast recovery area require the database to be restarted?

Answer: No, a restart is not required for this change.

g) Change the size of the Fast Recovery Area to 8 GB, by entering **8** into the “Flash Recovery Area Size” field and choosing **GB** from the pick-list next to the size field.

h) Optionally, click **Show SQL**, review the statement and click **Return**.

Show SQL Return

```
ALTER SYSTEM SET db_recovery_file_dest_size = 8589934592 SCOPE=BOTH
```

i) Click **Apply**.

4) Check how many members each redo log group has. Ensure that there are at least two redo log members in each group. One set of members should be stored in the fast recovery area.

a) Click **Server > Redo Log Groups**, and note how many members are in the “# of Members” column.

Practice 14-1: Configuring Your Database for Recovery (continued)

Answer: There are two members in each group.

Redo Log Groups

Object Type: Redo Log Group

Search
Enter an object name to filter the data that is displayed in your results set.
Object Name: Go

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode: Single

Select	Group	Status	# of Members	Archived	Size (KB)	Sequence	First Change#
<input checked="" type="radio"/>	1	Inactive	2	No	51200	127	3680499
<input type="radio"/>	2	Current	2	No	51200	128	3721156
<input type="radio"/>	3	Inactive	2	No	51200	126	3646837

b) Select one of your Redo Log Groups and click View to see where each member of that group is stored. You should see one member in the +DATA disk group and the second member in the +FRA disk group.

View Redo Log Group: 1

Actions: Clear logfile

Group # 1
File size 51200 KB
Status INACTIVE

Redo Log Members

File Name	File Directory
group_1.261.691672257	+DATA/orcl/online log/
group_1.257.691672265	+FRA/orcl/online log/

5) You notice that, for each log group, the Archived column has a value of No. This means that your database is not retaining copies of redo logs to use for database recovery, and in the event of a failure, you will lose all data since your last backup. Place your database in ARCHIVELOG mode, so that redo logs are archived.

Note: You must continue with step 5, so that your changes are applied.

- In Enterprise Manager, select Availability > Recovery Settings in the Setup section.
- In the Media Recovery region, select the **ARCHIVELOG Mode** check box. Also, verify that Log Archive Filename Format contains %t, %s, and %r.

Media Recovery

The database is currently in NOARCHIVELOG mode. In ARCHIVELOG mode, hot backups and recovery to the latest time are possible, but you must provide space for archived redo log files. If you change the database to ARCHIVELOG mode, you should perform a backup immediately. In NOARCHIVELOG mode, only cold backups are possible and data may be lost in the event of database corruption.

ARCHIVELOG Mode*

Log Archive Filename Format*

Number	Archived Redo Log Destination	Status	Type
1	USE_DB_RECOVERY_FILE_DEST	VALID	Local
<input type="button" value="Add Another Row"/>			

TIP It is recommended that archived redo log files be written to multiple locations spread across the different disks.
 TIP You can specify up to 10 archived redo log destinations.

Practice 14-1: Configuring Your Database for Recovery (continued)

- c) Notice the current configuration for the archive log destination is to USE_DB_RECOVERY_FILE_DEST, which points to the fast recovery area (+FRA).
Note: If you add archive log destinations, you must create the directory, if it does not already exist.
- d) Click **Apply**.
- e) When prompted whether you want to restart the database now, click **Yes**.
- f) Enter the host credentials to restart the database (oracle as the Username and Password) and then click **OK**.
- g) When asked to confirm, click **Yes** again.
- h) Should you receive an error during the shutdown and startup activity, click **OK** to acknowledge the error, and then click Refresh again. (You might have been simply faster than the database.)

6) Optionally, once your database has restarted, use SQL*Plus to check whether your database is in ARCHIVELOG mode. In a terminal window, log in to SQL*Plus as **SYSDBA** and run the **archive log list** command.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Sat Jul 11 10:16:40
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options

SQL> archive log list
Database log mode           Archive Mode
Automatic archival          Enabled
Archive destination          USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence   126
Next log sequence to archive 128
Current log sequence         128
SQL> exit
```

Practice 14-1: Configuring Your Database for Recovery (continued)

```
Disconnected from Oracle Database 11g Enterprise Edition
Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options
$
```

Now that your database is in ARCHIVELOG mode, it will continually archive a copy of each online redo log file before reusing it for additional redo data.

Note: Remember that this consumes space on the disk and that you must regularly back up older archive logs to some other storage.

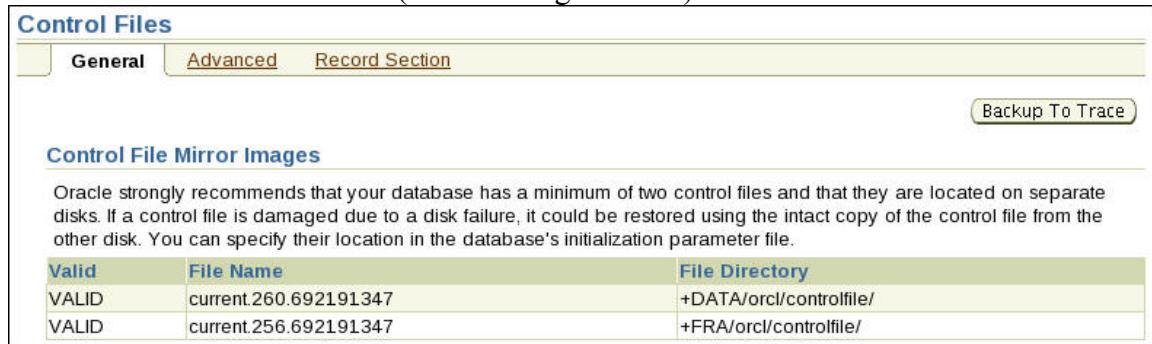
Practices for Lesson 15

Background: Now that your database is ready for production, it is time to start taking backups. Perform backup of the control file to trace, an immediate backup to disk, and schedule nightly backup jobs that repeat indefinitely.

Practice 15-1: Performing Database Backups

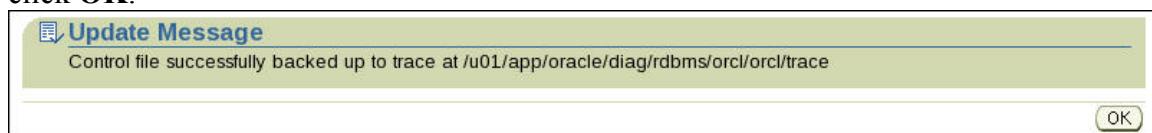
In this practice, you backup your control file to trace, perform an immediate backup to disk, and schedule a nightly backup job.

- 1) Perform a backup of the control file to trace.
 - a) Invoke Enterprise Manager as the **DBA1** user in the **SYSDBA** role for your **orcl** database.
 - b) Click **Server > Control Files** (in the Storage section).



Valid	File Name	File Directory
VALID	current.260.692191347	+DATA/orcl/controlfile/
VALID	current.256.692191347	+FRA/orcl/controlfile/

- c) Click **Backup to Trace**.
- d) When you receive the success message, note the trace directory location, and then click **OK**.



Update Message

Control file successfully backed up to trace at /u01/app/oracle/diag/rdbms/orcl/orcl/trace

OK

- e) Optionally, use a terminal window, logged in as the **oracle** user to view the trace file name at the end of the alert log by executing the following command:

```
cd /u01/app/oracle/diag/rdbms/orcl/orcl/trace
tail alert_orcl.log
```

The following output shows only the last few lines:

```
$ cd /u01/app/oracle/diag/rdbms/orcl/orcl/trace
$ tail alert_orcl.log
Sat Jul 11 09:10:03 2009
SMCO started with pid=23, OS id=9837
Sat Jul 11 09:46:31 2009
ALTER DATABASE BACKUP CONTROLFILE TO TRACE
Backup controlfile written to trace file
/u01/app/oracle/diag/rdbms/orcl/orcl/trace/orcl_ora_12190.trc
Completed: ALTER DATABASE BACKUP CONTROLFILE TO TRACE
Sat Jul 11 09:46:56 2009
ALTER DATABASE BACKUP CONTROLFILE TO TRACE
Backup controlfile written to trace file
/u01/app/oracle/diag/rdbms/orcl/orcl/trace/orcl_ora_12190.trc
Completed: ALTER DATABASE BACKUP CONTROLFILE TO TRACE
$
```

f) Optionally, to view size and usage of the different sections within the control file, click the Record Section tabbed page.

ORACLE Enterprise Manager 11g Database Control

Database Instance: orcl.oracle.com > Logged in As SYS

Control Files

Record Section

Control File Record Information

Type	Record Size	Records Total	Records Used
CKPT PROGRESS	8180	11	0
DATABASE	316	1	1
DATAFILE	520	100	7
FILENAME	524	2298	14
LOG HISTORY	56	292	5
REDO LOG	72	16	3
REDO THREAD	256	8	1
TABLESPACE	68	100	7

Your numbers could look different. For additional information, click Help in the upper-right corner of the page.

2) What is the difference between a backup set and an image copy?

Answer: A backup set contains data and archive log files packed in an Oracle proprietary format. Files must be extracted before use. Image copies are the equivalent of operating system file copies and can be used for restore operations immediately.

3) What is the destination of any disk backups that are done?

a) Navigate to the **Availability** page and click **Backup Settings**.

Backup Settings

Disk Settings

Parallelism: 1

Disk Backup Location: The fast recovery area is the current disk backup location. If you would like to override the disk backup location, specify an existing directory or diskgroup.

Disk Backup Type: **Backup Set**
An Oracle backup file format that allows for more efficient backups by interleaving multiple backup files into one output file.

Compressed Backup Set
An Oracle backup set in which the data is compressed to reduce its size.

Image Copy
A bit-by-bit copy of database files that can be used as-is to perform recovery.

b) Note the message under the Disk Backup Location that says the fast recovery area is the current disk backup location.

4) Establish the backup policy to automatically back up the SPFILE and control file.

a) Click the **Policy** tab under the Backup Settings heading.

b) Select “**Automatically backup the control file and server parameter file (SPFILE) with every backup and database structural change.**”

c) Scroll to the bottom and enter **oracle** and **oracle** for Host Credentials Username and Password for your server, and select “**Save as Preferred Credential.**”

d) Click **OK**.

5) Test making a backup to disk, as a backup set, with **oracle** for Host Credentials.

- Click the **Device** tab under the Backup Settings pages.
- Select **Backup Set** as your Disk Backup Type.
- Scroll to the bottom and ensure the Host Credentials are set to **oracle**.
- Scroll to the top of the page and click **Test Disk Backup**.
- A processing message appears. When the test finishes, and you see the “Disk Backup Test Successful!” message, scroll down to the bottom of the page and click **OK**.

6) Back up your entire database, with archive logs, while the database is open for user activity. This backup should be the base for an incremental backup strategy.

- Question:** What prerequisite *must* be met to create a valid backup of a database without shutting it down?
Answer: The database must be in ARCHIVELOG mode. Backups made with the database open, but not in ARCHIVELOG mode, cannot be used for recovery.
- Select **Availability > Schedule Backup** (in the Manage section).
If you find that the Oracle-Suggested Backup strategy fits your needs exactly, you would choose this option. For practice purposes, you will schedule a customized backup
- Select **Whole Database** as the object to be backed up.
- Confirm or enter **oracle** and **oracle** for Host Credentials Username and Password for your server.
- Click **Schedule Customized Backup**.

- f) On the Schedule Customized Backup: Options page, select **Full Backup** for your Backup Type, and select the “**Use as the base of an incremental backup strategy**” check box.
- g) Select **Online Backup** as Backup Mode.
- h) In the Advanced section, select “**Also back up all archived logs on disk**” and “**Delete all archived logs from disk after they are successfully backed up**”, and then click **Next** to continue.
- i) On the Schedule Customized Backup: Settings page, select **Disk** for your backup location. (Notice that your Disk Backup Location is retained and that you could override the current settings for a one-off backup. But do not click it this time.)

Schedule Customized Backup: Settings

Database	orcl.oracle.com	Cancel	Back	Step 2 of 4	Next
Backup Strategy	Customized Backup				
Object Type	Whole Database				
Select the destination media for this backup. You can also override the default backup settings.					
<input checked="" type="radio"/> Disk Disk Backup Location +FRA <input type="radio"/> Tape Media Management Vendor (MMV) Library Parameters Not specified					
View Default Settings Override Default Settings <small>Changed settings will only apply to the current backup.</small>					

- j) Click **Next**.
- k) Accept all the defaults on the Schedule Customized Backup: Schedule page and then click **Next** to continue.
Note: Schedule Type should be One Time (Immediately).
- l) On the Schedule Customized Backup: Review page, review the RMAN script, and then click **Submit Job**.

Schedule Customized Backup: Review

Database	orcl.oracle.com	Cancel	Edit RMAN Script	Back	Step 4 of 4	Submit Job
Backup Strategy	Customized Backup					
Object Type	Whole Database					
Settings						
Destination	Disk					
Backup Type	Use as the base of an incremental backup strategy					
Backup Mode	Online Backup					
Flash Recovery Area	+FRA					
RMAN Script						
The RMAN script below is generated based on previous input.						
<pre>backup incremental level 0 cumulative device type disk tag '%TAG' database; backup device type disk tag '%TAG' archivelog all not backed up delete all input;</pre>						

- m) Click **View Job** to monitor the status of the backup job. The time for this backup depends on your hardware and system resources.

n) Click your browser's **Refresh** or **Requery** button until the job is completed.

Execution: orcl.oracle.com

Page Refreshed Jul 11, 2009 10:35:15 AM GMT+07:00 [Delete Run](#) [Edit](#) [View Definition](#)

Summary

Status	Succeeded	Type	Database Backup
Scheduled	Jul 11, 2009 10:28:05 AM (UTC+07:00)	Owner	SYS
Started	Jul 11, 2009 10:28:05 AM (UTC+07:00)	Description	Whole Database Backup
Ended	Jul 11, 2009 10:34:32 AM (UTC+07:00)	Oracle SID	orcl
Elapsed Time	387 seconds	Backup Strategy	advanced
Notification	No	Version 10g or higher	YES
		Database Connect String	(DESCRIPTION=(ADDRESS_LIST=(ADDR...
		Database Name	ORCL
		Blackout	NO
		Encryption Mode	None
		Offline Backup	NO
		Backup Script	Show
		Oracle Home	/u01/app/oracle/product/11.2.0/d...
		Host Username	oracle
		Database Username	SYS
		Database Role	*****

Targets: Status: [All](#) [Go](#)

[Expand All](#) [Collapse All](#)

Name	Targets	Status	Started	Ended	Elapsed Time (seconds)
Execution: orcl.oracle.com	orcl.oracle.com	Succeeded	Jul 11, 2009 10:28:05 AM (UTC+07:00)	Jul 11, 2009 10:34:32 AM (UTC+07:00)	387
Step: Prebackup	orcl.oracle.com	Succeeded	Jul 11, 2009 10:28:25 AM (UTC+07:00)	Jul 11, 2009 10:28:26 AM (UTC+07:00)	1
Step: Backup	orcl.oracle.com	Succeeded	Jul 11, 2009 10:28:35 AM (UTC+07:00)	Jul 11, 2009 10:34:21 AM (UTC+07:00)	346
Step: Post Backup	orcl.oracle.com	Succeeded	Jul 11, 2009 10:34:30 AM (UTC+07:00)	Jul 11, 2009 10:34:32 AM (UTC+07:00)	2

7) Schedule nightly disk-based incremental online backups for your whole database, including archive logs. Have the archive logs deleted from disk after the backup is complete. Schedule it for execution at 11:00 PM. The schedule should be in effect indefinitely.

- In Enterprise Manager, select **Availability > Schedule Backup** (in the Manage section).
- Select **Whole Database** as the object to be backed up.
- Confirm or enter **oracle** and **oracle** for Host Credentials Username and Password for your server, and then click **Schedule Customized Backup**.
- On the Schedule Customized Backup: Options page, select **Incremental Backup** as your Backup Type.
- Select **Online Backup** as Backup Mode.
- In the Advanced region, select **“Also backup all archived logs on disk”** and **“Delete all archived logs from disk after they are successfully backed up”**, and then click **Next** to continue.
- On the Schedule Customized Backup: Settings page, select **Disk** as your backup location, and then click **Next** to continue.
- On the Schedule Customized Backup: Schedule page, change Job Name to **Nightly_Backup** and accept the default value for Job Description.
- Select **Repeating** in the Schedule region. Notice how additional context-sensitive details are displayed.

j) Select **By Days** from the Frequency Type drop-down list, enter **1** in the Repeat Every field, confirm that Indefinite is selected as the Repeat Until value, and enter **11:00 PM** as Start Time.

Schedule Customized Backup: Schedule

Database: **orcl.oracle.com**
 Backup Strategy: **Customized Backup**
 Object Type: **Whole Database**

Job

* Job Name: **Nightly_Backup**
 Job Description: **Whole Database Backup**

Schedule

Type: One Time (Immediately) One Time (Later) Repeating

Frequency Type: **By Days**

Repeat Every: **1** Days

Time Zone: **(UTC-06:00) US Central Time (CST)**

Start Date: **Jul 11, 2009**

Start Time: **11:00** AM PM

Repeat Until: Indefinite
 Specified Date
 Date: **Jul 11, 2009**
(example: Jul 11, 2009)

Time: **11:00** AM PM

k) Click **Next** to continue.

l) On the Schedule Customized Backup: Review page, review your Settings and RMAN script.

Schedule Customized Backup: Review

Database: **orcl.oracle.com**
 Backup Strategy: **Customized Backup**
 Object Type: **Whole Database**

Settings

Destination	Disk
Backup Type	Incremental Backup
Backup Mode	Online Backup
Flash Recovery Area	+FRA

RMAN Script

The RMAN script below is generated based on previous input.

```
backup incremental level 1 cumulative device type disk tag '%TAG' database;
backup device type disk tag '%TAG' archivelog all not backed up delete all input;
```

m) Click **Submit Job**, and then click **OK**.

n) Click **Jobs** on the Availability page in the Related Links section to see the scheduled job in the Job Activity list.

View Runs

View Results	Edit	Create Like	Copy To Library	Suspend	Resume	Stop	Delete	Create Job	OS Command	Go
Select	Name	Status (Executions)	Scheduled	Targets	Target Type	Owner	Job Type			
<input checked="" type="radio"/>	NIGHTLY_BACKUP	1 Scheduled	Jul 11, 2009 11:00:00 PM CDT	orcl.oracle.com	Database Instance	SYS	Database Backup			
<input type="radio"/>	BACKUP_ORCL.ORACLE.COM_000003	1 Succeeded	Jul 11, 2009 10:28:05 AM (UTC+07:00)	orcl.oracle.com	Database Instance	SYS	Database Backup			
<input type="radio"/>	REORGANIZE_ORCL.ORACLE.COM_1	1 Succeeded	Jul 9, 2009 10:52:46 AM (UTC+07:00)	orcl.oracle.com	Database Instance	SYS	Reorganize			

Practices for Lesson 16

Background: Many failures of the Oracle database can be traced to some sort of media failure, such as disk or controller failure. In this practice, you encounter a number of problems from which you need to recover the database.

- Recover from the loss of a data file
- Recover from the loss of a file in the SYSTEM tablespace
- Recover from the loss of a control file

SQL script files are provided for you in the /home/oracle/labs directory. If needed, use the appendixes for Linux and for SQL syntax. *After you set up a failure with a SQL script, you must complete the recovery before continuing with any other practice.*

Note: Your system may have different OS file names than shown here. Your output might look different. (To conserve space, blank lines have been removed.)

Practice 16-1: Preparing Practice Environment

Before beginning one of the recovery scenarios, you need to run a script that will prepare the environment for the remaining recovery practices.

- 1) Before setting up an individual problem, you must navigate to your **labs** directory and (in SQL*Plus) execute the **lab_16_01_01.sql** script as the **SYS** user. This script prepares some procedures to be called by the rest of this practice.

```
$ cd /home/oracle/labs
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Sun Jul 12 23:29:58
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options

SQL> @lab_16_01_01.sql
Connected.

Java created.

Procedure created.

PL/SQL procedure successfully completed.

Synonym created.

Grant succeeded.

SQL>
```

Practice 16-2: Recovering from the Loss of a Data File

In this practice, you experience the loss of an application data file. You then go through the steps to recover from this loss.

- 1) In a SQL*Plus session, as the SYS user, execute the `lab_16_02_01.sql` script from your `labs` directory. This script deletes one of your application data files.

Note: Remember to use `oraenv` to set your environment to the `orcl` database if you have not already done so in your terminal window.

```
$ sqlplus / as sysdba

SQL*Plus: Release 11.1.0.6.0 - Production on Sun Aug 31
10:45:19 2008

Copyright (c) 1982, 2007, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 -
Production
With the Partitioning, OLAP, Data Mining and Real Application
Testing options

SQL> @lab_16_02_01.sql

PL/SQL procedure successfully completed.

>Data file deleted. Database should be open."
SQL>
```

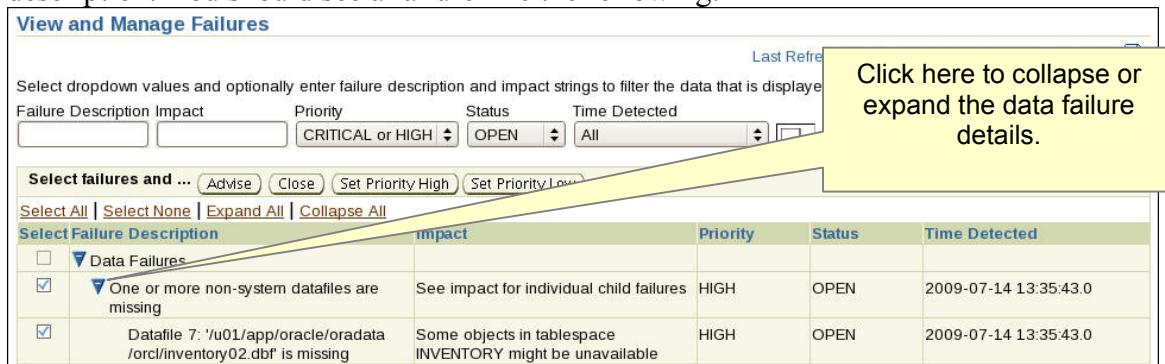
- 2) The Help desk has received a call from a user who is unable to access the `COUNTRIES` table in the `HR` application schema. Count the rows in the table to determine whether there is a problem.
Note: If you do not see the data file error, try doing an update to the `INVENTORY.PRODUCT_MASTER` table from another SQL*Plus session. It may take a little while before the Oracle database realizes that the file is missing. If you do not want to wait you may skip this step and move on with the next step.

```
SQL> select count(*) from INVENTORY.PRODUCT_MASTER;
select count(*) from INVENTORY.PRODUCT_MASTER
*
ERROR at line 1:
ORA-00376: file 7 cannot be read at this time
ORA-01110: data file 7:
'/u01/app/oracle/oradata/orcl/inventory02.dbf'
SQL>
```

- 3) Troubleshoot and recover as necessary. The error message suggests that the `inventory02.dbf` data file is corrupt or missing.
 - a) In Enterprise Manager, on the Home page, look in the Alerts section and notice the Data Failure alert.

Practice 16-2: Recovering from the Loss of a Data File (continued)

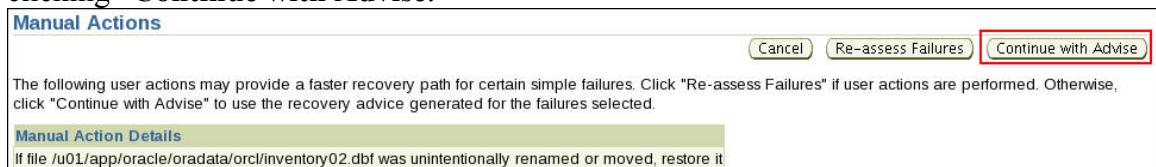
- b) Click Availability > Perform Recovery.
- c) Click Advise and Recover.
- d) On the View and Manage Failures page, click the plus (+) icon under the failure description. You should see a failure like the following:



Select Failure Description	Impact	Priority	Status	Time Detected
<input type="checkbox"/> Data Failures				
<input checked="" type="checkbox"/> One or more non-system datafiles are missing	See impact for individual child failures	HIGH	OPEN	2009-07-14 13:35:43.0
<input checked="" type="checkbox"/> Datafile 7: '/u01/app/oracle/oradata/orcl/inventory02.dbf' is missing	Some objects in tablespace INVENTORY might be unavailable	HIGH	OPEN	2009-07-14 13:35:43.0

Note: If you do not see the nonsystem datafile failure, keep refreshing the page until it shows up.

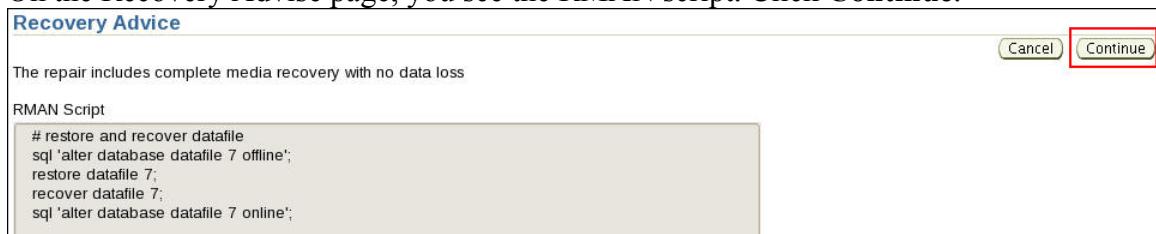
- e) With the failures selected, click **Advise**.
- f) Because the file was not just renamed or moved, but deleted, you continue by clicking **“Continue with Advise.”**



The following user actions may provide a faster recovery path for certain simple failures. Click "Re-assess Failures" if user actions are performed. Otherwise, click "Continue with Advise" to use the recovery advice generated for the failures selected.

Manual Action Details
If file /u01/app/oracle/oradata/orcl/inventory02.dbf was unintentionally renamed or moved, restore it

- g) On the Recovery Advise page, you see the RMAN script. Click **Continue**.

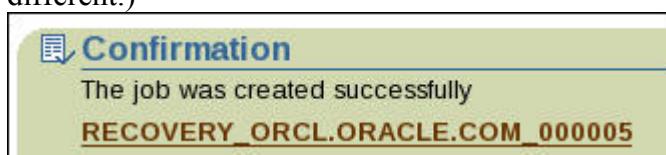


The repair includes complete media recovery with no data loss

RMAN Script

```
# restore and recover datafile
sql 'alter database datafile 7 offline';
restore datafile 7;
recover datafile 7;
sql 'alter database datafile 7 online';
```

- h) On the Review page, you see the failure and the suggested solution. Click **“Submit Recovery Job.”**
- i) A Processing window appears, followed by the Job Activity page. You should see a message that the job was successfully created. (Your link name is probably different.)



Confirmation

The job was created successfully

[RECOVERY_ORCL.ORACLE.COM_000005](#)

- j) Click the job name link.

Practice 16-2: Recovering from the Loss of a Data File (continued)

k) On the Job Run page, check the Status in the Summary section. If it is Running, use your browser's **Refresh** or Requery button until the job is completed.

Summary

Status	Succeeded
Scheduled	Jul 14, 2009 1:44:35 PM (UTC+07:00)
Started	Jul 14, 2009 1:44:35 PM (UTC+07:00)
Ended	Jul 14, 2009 1:45:18 PM (UTC+07:00)
Elapsed Time	42 seconds
Notification	No

l) In your SQL*Plus session, verify that the INVENTORY.PRODUCT_MASTER table is now accessible.

Note: Your count value may be different than the one shown here.

```
SQL> select count(*) from INVENTORY.PRODUCT_MASTER;
```

```
-----  
COUNT(*)  
-----  
217368
```

m) Now that recovery is complete, in Enterprise Manager, navigate to the database Home page and clear the alert.

- Click the link in the Message column for the Data Failure alert.
- Click **Clear Open Alert**.

Practice 16-3: Recovering from the Loss of a File in the SYSTEM Tablespace

In this practice, your system experiences the loss of a file in the SYSTEM tablespace. You then go through the steps to recover from this loss.

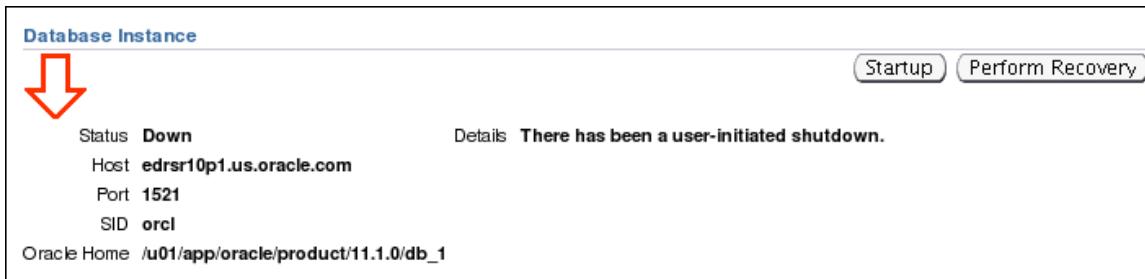
- 1) Why is recovery from the loss of a system data file or a data file belonging to an undo tablespace different from recovering an application data file?

Answer: Because recovery of system or undo data files must be done with the database closed, whereas recovery of an application data file can be done with the database open and available to users

- 2) As the `oracle` OS user, execute the `lab_16_03_02.sh` script in your `labs` directory. This script deletes the system data file.

```
$ ./lab_16_03_02.sh
```

- 3) In Enterprise Manager, review the Database home page. If you see a message that says the connection was refused, try re-entering the EM home page URL in the browser. You may need to try several times before you see the Database home page.



The screenshot shows the Oracle Database Home page. At the top, it says "Database Instance". Below that, there is a status indicator with a red arrow pointing to the word "Down". To the right of the status are two buttons: "Startup" and "Perform Recovery". Under the status, there are details: "Host edrsr10p1.us.oracle.com", "Port 1521", "SID orcl", and "Oracle Home /u01/app/oracle/product/11.1.0/db_1".

- 4) The database is shut down. Attempt to start your database.
 - a) Click **Startup** to try to open it.
 - b) On the "Startup/Shutdown:Specify Host and Target Database Credentials" page, enter `oracle` and `oracle` as Host Credentials. Click **OK**.
 - c) On the Select Startup Type page, leave the default selection and click **OK**.
 - d) On the Startup/Shutdown:Confirmation page, click **Yes**.

Practice 16-3: Recovering from the Loss of a File in the SYSTEM Tablespace (continued)

e) A Srvctl Error appears saying that data file 1 cannot be identified. Click the **Database** tab to go back to your database Home page.

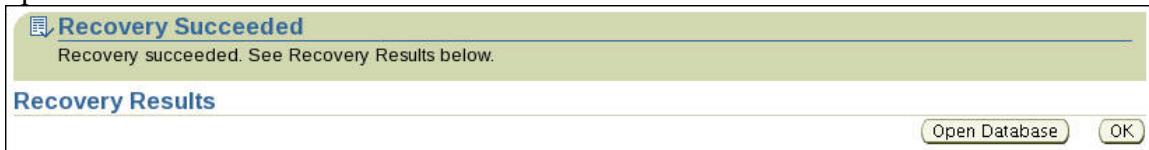
The screenshot shows the Oracle Enterprise Manager 11g Database Control interface. At the top, there is a banner with the text: "An error occurred when executing the operation. Ensure that CRS processes and SRVCTL are functioning properly. Refresh the page to see the current status. Refer to the following error: startup open PRCR-1079 : Failed to start resource ora.orcl.db ORA-01157: cannot identify/lock data file 1 - see DBWR trace file ORA-01110: data file 1: '+DATA/orcl/datafile /system.256.692191147' CRS-2674: Start of 'ora.orcl.db' on 'edrsr12p1' failed". Below this, the "Startup/Shutdown: Select Operation" page is displayed. It shows a table with one row for the instance "orcl". The "Status" column for "orcl" is red with a downward arrow, indicating it is down. The "Host Name" column is "n/a".

5) Click **Perform Recovery**.

- a) Enter **oracle** and **oracle** as Host Credentials, and click **Continue**.
- b) Your database is currently down, so it needs to be started in a mounted state to perform the recovery. Click the **Start up the Database** link.
 - i) You are prompted again for the host credentials, enter **oracle** and **oracle** and click **OK**.
 - ii) On the Select Startup Type page leave the default selection and click **OK**.
 - iii) Click **Advanced Options** on the Startup/Shutdown: Confirmation page.
 - iv) Select **Mount the database** for the Startup mode and click **OK**.
 - v) Click **Yes** back on the Startup/Shutdown: Confirmation page.
 - vi) Startup is Successful. Click the **Database** tab.
- c) Notice that your database is now in a mounted state. Click **Perform Recovery** again.
- d) In the Host Credentials section, enter **oracle** for the username and password. Then click **Continue**.
- e) On the Database Login page, enter **dba1** for the username, **oracle_4U** for the password, and **SYSDBA** for Connect As. Then click **Login**.
- f) On the Perform Recovery page, click **Advise and Recover**.
- g) On the View and Manage Failures page, review the description of the failure. Ensure the System datafile failure is selected and then click **Advise**.
- h) On the Manual Actions page, click **Continue with Advise**.
- i) On the Recovery Advice page, review the RMAN script and click **Continue**.
- j) On the Review page, click **Execute Recovery**.

Practice 16-3: Recovering from the Loss of a File in the SYSTEM Tablespace (continued)

k) A processing page appears, followed by the Perform Recovery: Result page. The duration of this operation depends on your system resources. The recovery operation should be successful.



- l) On the Perform Recovery: Result page, click **Open Database**.
- m) After you see the success message, click **OK**.
- n) Verify that the database is open and operating normally, by logging in to EM as the **DBA1** user as **SYSDBA**, and reviewing the Database home page.
- o) Now that recovery is complete, clear the alert.
 - i) Click the link in the Message column for the Data Failure alert.
 - ii) Click **Clear Open Alert**.

Practice 16-4: Recovering from the Loss of a Control File

In this practice, your system experiences the loss of a control file. To introduce the RMAN command-line environment, this practice guides you through a manual recovery process.

- 1) In a terminal window as the `oracle` user, run the `lab_16_04_01.sh` script. This script deletes one of your control files.

```
./lab_16_04_01.sh
```

- 2) The Help desk begins receiving calls saying that the database appears to be down. Troubleshoot and recover as necessary. Use `SRVCTL` to try to start up the database.
 - a) In a terminal window, ensure that your environment is configured for your `orcl` database environment using `oraenv`.
 - b) Issue the following `SRVCTL` command to attempt to start up the database:
`srvctl start database -d orcl -o open`

```
$ srvctl start database -d orcl -o open
PRCR-1079 : Failed to start resource ora.orcl.db
ORA-00205: error in identifying control file, check alert log
for more info
CRS-2674: Start of 'ora.orcl.db' on 'edrsr12p1' failed
```

- 3) Further investigate this failure by looking at your `alert_orcl.log` file (located in `/u01/app/oracle/diag/rdbms/orcl/orcl/trace` directory): In a terminal window, enter the following to see the last 20 lines of your `alert_orcl.log` file.

```
$ tail -20 alert_orcl.log
ALTER DATABASE MOUNT
NOTE: Loaded library:
/opt/oracle/extapi/32/asm/orcl/1/libasm.so
NOTE: Loaded library: System
SUCCESS: diskgroup DATA was mounted
SUCCESS: diskgroup FRA was mounted
ORA-00210: cannot open the specified control file
ORA-00202: control file:
'+FRA/orcl/controlfile/current.256.695531381'
ORA-17503: ksfdopn:2 Failed to open file
+FRA/orcl/controlfile/current.256.695531381
ORA-15012: ASM file
'+FRA/orcl/controlfile/current.256.695531381' does not exist
NOTE: dependency between database orcl and diskgroup resource
ora.DATA.dg is established
Sat Aug 22 03:18:05 2009
Checker run found 1 new persistent data failures
ORA-205 signalled during: ALTER DATABASE MOUNT...
Sat Aug 22 03:18:08 2009
Shutting down instance (abort)
```

Practice 16-4: Recovering from the Loss of a Control File (continued)

```
License high water mark = 1
USER (ospid: 17765) : terminating the instance
Instance terminated by USER, pid = 17765
Sat Aug 22 03:18:08 2009
Instance shutdown complete
$
```

- 4) Notice that the missing control file is the one from your **+FRA** disk group. You know you also have a control file on the **+DATA** disk group. You can perform a recovery by restoring from the control file that is in the **+DATA** disk group, but you need to know the file name. Using **asmcmd**, determine the name of the control file in the **+DATA** disk group.
 - a) Open a new terminal window and set your environment for your **+ASM** instance using **oraenv**.

```
$ . oraenv
ORACLE_SID = [oracle] ? +ASM
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
```

- b) Start **asmcmd** and use the **ls** command to determine the name of the control file in the **+DATA** disk group (this file will be in the **+data/orcl/controlfile** directory).

```
$ asmcmd
ASMCMD> ls +data/orcl/controlfile
Current.260.695209463
ASMCMD>
```

- c) Make a note of this name along with its full path because you will need this information for the next step.
- 5) In another terminal window, connect to **RMAN** and use the following command to restore your control file:
restore controlfile from '+DATA/orcl/controlfile/yourcontrolfilename';
Then mount and open your database.
 - a) Set your environment for your **orcl** database using **oraenv** and then connect to **RMAN**.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$ rman target /
Recovery Manager: Release 11.2.0.2.0 - Production on Sat Aug
22 03:27:29 2009
```

Practice 16-4: Recovering from the Loss of a Control File (continued)

```
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.
```

```
connected to target database (not started)
```

```
RMAN>
```

b) First, you need to put your database into NOMOUNT mode.

```
RMAN > startup nomount

Oracle instance started

Total System Global Area      577511424 bytes

Fixed Size                  1337984 bytes
Variable Size                469763456 bytes
Database Buffers             100663296 bytes
Redo Buffers                 5746688 bytes

RMAN>
```

c) Restore the control file from the existing control file on the +DATA disk group.

Note: Use the file name determined in step 4.

```
RMAN> restore controlfile from
' +DATA/orcl/controlfile/current.260.695209463';

Starting restore at 22-AUG-09
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=23 device type=DISK

channel ORA_DISK_1: copied control file copy
output file name=+DATA/orcl/controlfile/current.260.695209463
output file name=+FRA/orcl/controlfile/current.256.695532623
Finished restore at 22-AUG-09

RMAN>
```

d) Mount your database.

```
RMAN> alter database mount;

database mounted
```

e) Open your database.

```
RMAN> alter database open;

database open
```

***Practice 16-4: Recovering from the Loss of a Control File
(continued)***

- 6) Open a browser and log in to Enterprise Manager as the **dba1** user. On the Database home page, confirm that your database is again up and running.
- 7) Exit all sessions and close all windows.

Practices for Lesson 17

Background: In the recent past, you received a number of questions about the HR schema. To analyze them without interfering in daily activities, you decide to use the Data Pump Wizard to export the HR schema to file. When you perform the export, you are not sure into which database you will be importing this schema.

In the end, you learn that the only database for which management approves an import is the `orc1` database. So you perform the import with the Data Pump Wizard, remapping the HR schema to DBA1 schema.

Then you receive two data load requests for which you decide to use SQL*Loader.

Practice 17-1: Moving Data Using Data Pump

In this practice, you first grant the DBA1 user the privileges necessary to provide access to the DATA_PUMP_DIR directory. You then export the HR schema so that you can then import the tables that you want into the DBA1 schema. In the practice, you import only the EMPLOYEES table at this time.

- 1) First, you need to grant to the DBA1 user the appropriate privileges on the DATA_PUMP_DIR directory and create the users and roles required for this practice. A script exists that performs all the steps required to configure your environment for this practice.
 - a) Review the **lab_17_01_01.sql** script, which grants the DBA1 user privileges on the DATA_PUMP_DIR directory and performs other configurations to your environment, by executing the following in your **labs** directory:

```
$ cat lab_17_01_01.sql
```

- b) The **lab_17_01_01.sh** script calls the **lab_17_01_01.sql** script. Execute the **lab_17_01_01.sh** script now:

```
$ ./lab_17_01_01.sh
Create HR_TEST tablespace, HR_TEST_ROLE role, the HR_TEST
users
```

Tablespace created.

User created.

Role created.

Grant succeeded.

Grant succeeded.

Table altered.

Grant succeeded.

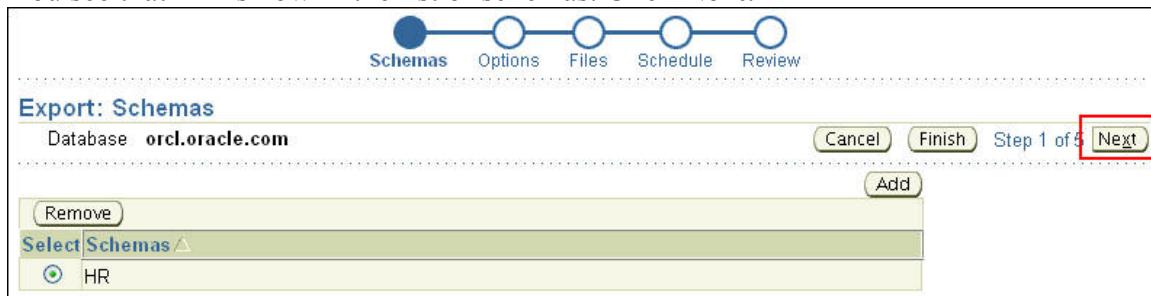
Grant succeeded.

```
You may now login as: hr_test/oracle_4U
spool off
$
```

Practice 17-1: Moving Data Using Data Pump (continued)

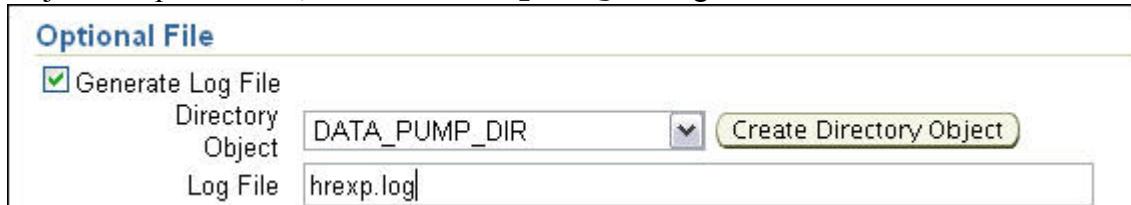
2) Log in to Enterprise Manager as the **DBA1** user in the **Normal** role and export the **HR** schema.

- Invoke Enterprise Manager as the **DBA1** user as the **Normal** role for your **orcl** database. The **Connect As** setting should be **Normal**.
- Select **Data Movement** > **Move Row Data** > **Export to Export Files**.
- Select **Schemas**, enter **oracle** as Username and Password, select **Save as Preferred Credential**, and then click **Continue**.
- On the Export: Schemas page, click **Add**, select the **HR** schema, and then click the **Select** button.
- You see that **HR** is now in the list of schemas. Click **Next**.



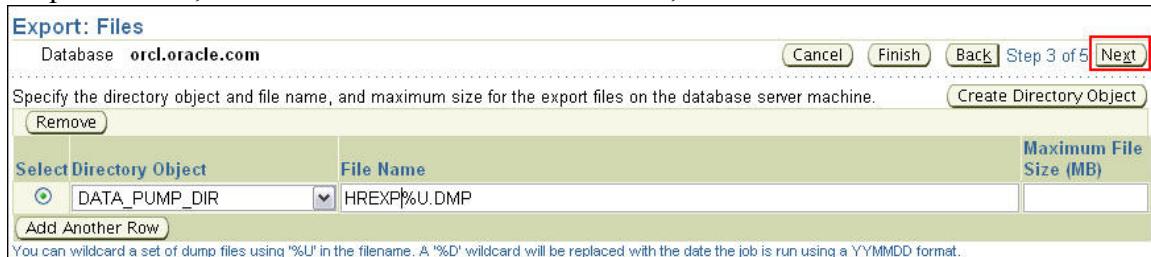
The screenshot shows the 'Export: Schemas' step of the Data Pump wizard. The 'Schemas' tab is active. A list box displays 'Select Schemas' with 'HR' selected. The 'Next' button is highlighted with a red box.

- On the Export: Options page, select **DATA_PUMP_DIR** from the Directory Objects drop-down list, and enter **hrexp.log** as Log File.



The screenshot shows the 'Optional File' step of the Data Pump wizard. The 'Generate Log File' checkbox is checked. The 'Directory Object' dropdown is set to 'DATA_PUMP_DIR' and the 'Log File' field contains 'hrexp.log'. The 'Next' button is highlighted with a red box.

- Review Advanced Options (but do not change), and then click **Next**.
- On the Export: Files page, select **DATA_PUMP_DIR** from the Directory Object drop-down list, enter **HREXP%U.DMP** as File Name, and then click **Next**.



The screenshot shows the 'Export: Files' step of the Data Pump wizard. The 'Select Directory Object' dropdown is set to 'DATA_PUMP_DIR' and the 'File Name' field contains 'HREXP%U.DMP'. The 'Next' button is highlighted with a red box.

Practice 17-1: Moving Data Using Data Pump (continued)

i) On the Export: Schedule page, enter **hrexp** as Job Name and **Export HR schema** as Description, accept the immediate job start time, and then click **Next**.

Export: Schedule

Database **orcl.oracle.com**

Specify a name and description for the export job. Specify a date to start the job.

Job Parameters

Job Name **hrexp**
Description **Export HR schema**

Job Schedule

Time Zone **(UTC-06:00) US Central Time**

Start

Immediately

Later

j) On the Export: Review page, click **Show PL/SQL** and review the PL/SQL that the Export Wizard helped you to create.

k) Click **Submit Job** to submit the job.

l) Click the link to the HREXP job to monitor the progress. Refresh your browser until you see that the job has successfully completed then move on to the next step.

Summary

Status	Succeeded
Scheduled	Jul 14, 2009 2:44:00 PM (UTC+07:00)
Started	Jul 14, 2009 2:44:00 PM (UTC+07:00)
Ended	Jul 14, 2009 2:45:45 PM (UTC+07:00)
Elapsed Time	105 seconds
Notification	No

3) Now, import the **EMPLOYEES** table from the exported **HR** schema into the **DBA1** schema. To get a feeling for the command-line interface, you can use the **impdp** utility from the command line to import the **EMPLOYEES** table into the **DBA1** user schema.

a) Ensure that your environment is configured for the **orcl** database by running **oraenv**.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$
```

b) Enter the following entire command string. Do not press [Enter] before reaching the end of the command:

```
impdp dba1/oracle_4U DIRECTORY=data_pump_dir
DUMPFILE=HREXP01.DMP REMAP_SCHEMA=hr:dba1
TABLES=hr.employees LOGFILE=empimport.log
```

Practice 17-1: Moving Data Using Data Pump (continued)

```
$ impdp dba1/oracle_4U DIRECTORY=data_pump_dir
DUMPFILE=HREXP01.DMP REMAP_SCHEMA=hr:dba1 TABLES=hr.employees
LOGFILE=empimport.log

Import: Release 11.2.0.1.0 - Production on Thu Jul 16 00:46:18
2009

Copyright (c) 1982, 2009, Oracle and/or its affiliates. All
rights reserved.

Connected to: Oracle Database 11g Enterprise Edition Release
11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options
Master table "DBA1"."SYS_IMPORT_TABLE_01" successfully
loaded/unloaded
Starting "DBA1"."SYS_IMPORT_TABLE_01": dba1/*********
DIRECTORY=data_pump_dir DUMPFILE=HREXP01.DMP
REMAP_SCHEMA=hr:dba1 TABLES=hr.employees LOGFILE=empimport.log
Processing object type SCHEMA_EXPORT/TABLE/TABLE
Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA
. . imported "DBA1"."EMPLOYEES" 16.81
KB      107 rows
Processing object type
SCHEMA_EXPORT/TABLE/GRANT/OWNER_GRANT/OBJECT_GRANT
Processing object type SCHEMA_EXPORT/TABLE/INDEX/INDEX
Processing object type
SCHEMA_EXPORT/TABLE/CONSTRAINT/CONSTRAINT
Processing object type
SCHEMA_EXPORT/TABLE/INDEX/STATISTICS/INDEX_STATISTICS
Processing object type SCHEMA_EXPORT/TABLE/COMMENT
Processing object type
SCHEMA_EXPORT/TABLE/CONSTRAINT/REF_CONSTRAINT
ORA-39083: Object type REF_CONSTRAINT failed to create with
error:
ORA-00942: table or view does not exist
Failing sql is:
ALTER TABLE "DBA1"."EMPLOYEES" ADD CONSTRAINT "EMP_JOB_FK"
FOREIGN KEY ("JOB_ID") REFERENCES "DBA1"."JOBS" ("JOB_ID")
ENABLE

ORA-39083: Object type REF_CONSTRAINT failed to create with
error:
ORA-00942: table or view does not exist
Failing sql is:
ALTER TABLE "DBA1"."EMPLOYEES" ADD CONSTRAINT "EMP_DEPT_FK"
FOREIGN KEY ("DEPARTMENT_ID") REFERENCES "DBA1"."DEPARTMENTS"
("DEPARTMENT_ID") ENABLE

Processing object type SCHEMA_EXPORT/TABLE/TRIGGER
```

Practice 17-1: Moving Data Using Data Pump (continued)

```
ORA-39082: Object type TRIGGER: "DBA1"."SECURE_EMPLOYEES"
created with compilation warnings
ORA-39082: Object type TRIGGER: "DBA1"."SECURE_EMPLOYEES"
created with compilation warnings
ORA-39082: Object type TRIGGER: "DBA1"."UPDATE_JOB_HISTORY"
created with compilation warnings
ORA-39082: Object type TRIGGER: "DBA1"."UPDATE_JOB_HISTORY"
created with compilation warnings
Processing object type
SCHEMA_EXPORT/TABLE/STATISTICS/TABLE_STATISTICS
Job "DBA1"."SYS_IMPORT_TABLE_01" completed with 6 error(s) at
00:46:45
```

Note: You may see errors on constraints and triggers not being created because only the EMPLOYEES table is imported and not the other objects in the schema. These errors are expected.

- c) You can also verify that the import succeeded by viewing the log file.

```
$ cat /u01/app/oracle/admin/orcl/dpdump/empimport.log
```

- 4) Confirm that the EMPLOYEES table has been loaded into the DBA1 schema by logging in to SQL*Plus as the **DBA1** user and selecting data from the **EMPLOYEES** table.
 - a) Log in to SQL*Plus as the **DBA1** user.
Note: Remember to use oraenv to set your environment to the **orcl** database if you have not already done so in your terminal window.

```
$ sqlplus dba1

SQL*Plus: Release 11.1.0.6.0 - Production on Sun Aug 31
16:54:32 2008

Copyright (c) 1982, 2007, Oracle. All rights reserved.

Enter Password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.0.6.0 -
Production
With the Partitioning, OLAP, Data Mining and Real Application
Testing options

SQL>
```

- b) Select a count of the rows from the **EMPLOYEES** table in the **DBA1** schema, for verification of the import.

Practice 17-1: Moving Data Using Data Pump (continued)

```
SQL> select count(*) from employees;  
  
  COUNT(*)  
-----  
      107  
SQL>
```

Practice 17-2: Loading Data Using SQL*Loader

In this practice, you load data into the **PRODUCT_MASTER** table by using SQL*Loader via Enterprise Manager Database Control. Data and control files are provided.

- 1) As the **DBA1** user, use Enterprise Manager to load the **lab_17_02_01.dat** data file. This data file contains rows of data for the **PRODUCT_MASTER** table. The **lab_17_02_01.ctl** file is the control file for this load.
Optionally, view the **lab_17_02_01.dat** and **lab_17_02_01.ctl** files to learn more about their structure before going further.
 - a) Invoke Enterprise Manager as the **DBA1** user as the **Normal** role for your **orcl** database.
 - b) Select **Data Movement** > **Move Row Data** > **Load Data from User Files**.
 - c) Click **Use Existing Control File**. If not already entered, enter **oracle** as Username and as Password, click **Save as Preferred Credential**, and then click **Continue**.

Load Data: Generate Or Use Existing Control File

Database **orcl.oracle.com** **Cancel** **Continue**

Automatically Generate Control File
A control file will be generated after you define the structure of the data file.

Use Existing Control File
Allows you to use an existing control file that defines the structure of the data file.

Host Credentials

* Username **oracle**
* Password *********
 Save as Preferred Credential

- d) On the Load Data: Control File page, enter **/home/oracle/labs/lab_17_02_01.ctl** as the control file name and path, or use the flashlight icon to select this control file. Click **Next**.

Load Data: Control File

Database **orcl.oracle.com** **Cancel** **Finish** **Step 1 of 6** **Next**

A control file is used to describe what will be loaded and how. Specify the full path and name of the control file on the database server machine.

/home/oracle/labs/lab_17_02_01.ctl

- e) On the Load Data: Data File page, click **Provide the full path and name on the database server machine** and enter **/home/oracle/labs/lab_17_02_01.dat** as the data file name and path, or use the flashlight icon to select this data file. Click **Next**.

Load Data: Data File

Database **orcl.oracle.com** **Cancel** **Finish** **Back** **Step 2 of 6** **Next**

How would you like to specify the file containing the data?

The data file is specified in the control file

Provide the full path and name on the database server machine
/home/oracle/labs/lab_17_02_01.dat

Practice 17-2: Loading Data Using SQL*Loader (continued)

f) On the Load Data: Load Method page, select **Conventional Path**, and then click **Next**.

Load Data: Load Method

Database **orcl.oracle.com**

Conventional Path
Runs SQL INSERT statements to load data into database tables. Use this option if none of the choices below are appropriate.

Direct Path (faster)
Formats data blocks and writes them directly to the database files. No writing to clustered tables. No other writing to destination tables in progress. No SQL in the control file.

Parallel Direct Path (fastest)
Writes data into the same table or into the same partition table in concurrent sessions. Data is only appended. Triggers and constraints are disabled. No indexes maintained.

Cancel **Finish** **Back** **Step 3 of 6** **Next**

g) On the Load Data: Options page, accept all defaults, but enter **/home/oracle/labs/lab_17_02_01.log** as the log file name and path. Review the advanced options if you want, but do not change any, and then click **Next**.

Optional Files

Generate bad file where records which contain errors are stored
Bad File

Generate discard file where rejected and uninserted records are stored
Discard File

Generate log file where logging information is to be stored
Log File **/home/oracle/labs/lab_17_02_01.log** 

Show Advanced Options

Cancel **Finish** **Back** **Step 4 of 6** **Next**

h) On the Load Data: Schedule page, enter **lab_17_02_01** as Job Name and **Load data into the PRODUCT_MASTER table** as Description. Let the job start immediately, and then click **Next**.

Load Data: Schedule

Database **orcl.oracle.com**

Specify a name and description for the load data job. Specify a date to start the job.

Job Parameters

Job Name **lab_17_02_01**
Description **Load data into the PRODUCT_MASTER table**

Job Schedule

Time Zone **(UTC-06:00) US Central Time**

Start

Immediately
 Later

Cancel **Back** **Step 5 of 6** **Next**

i) On the Load Data: Review page, review the loading information and parameters, and then click **Submit Job**.

Practice 17-2: Loading Data Using SQL*Loader (continued)

j) Click the link to the `LAB_17_02_01` job to monitor the progress. After the job shows as successfully completed, move on to the next step.

Summary	
Status	Succeeded
Scheduled	Jul 14, 2009 2:55:29 PM (UTC+07:00)
Started	Jul 14, 2009 2:55:29 PM (UTC+07:00)
Ended	Jul 14, 2009 2:55:37 PM (UTC+07:00)
Elapsed Time	8 seconds
Notification	No

k) Confirm your results by viewing your `lab_17_02_01.log` file in your `/home/oracle/labs` directory.

2) As the **INVENTORY** user, load data into the **PRODUCT_ON_HAND** table by using SQL*Loader command line. The `lab_17_02_02.dat` data file contains rows of data for the **PRODUCT_ON_HAND** table. The `lab_17_02_02.ctl` file is the control file for this load.

Optionally, view the `lab_17_02_02.dat` and `lab_17_02_02.ctl` files to learn more about their structure before going further.

- Open a terminal window and navigate to the `/home/oracle/labs` directory.
- Ensure that your environment is configured for the `orcl` database by running `oraenv`.

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base for
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1 is
/u01/app/oracle
$
```

c) Enter the following SQL*Loader command (in continuation, without pressing [Enter] before reaching the end of the command):

```
sqlldr userid=inventory/oracle_4U control=lab_17_02_02.ctl
log=lab_17_02_02.log data=lab_17_02_02.dat
```

```
$ sqlldr userid=inventory/oracle_4U control=lab_17_02_02.ctl
log=lab_17_02_02.log data=lab_17_02_02.dat

SQL*Loader: Release 11.2.0.1.0 - Production on Tue Jul 14
14:56:44 2009

Copyright (c) 1982, 2009, Oracle and/or its affiliates. All
rights reserved.

Commit point reached - logical record count 64
Commit point reached - logical record count 83
$
```

Practice 17-2: Loading Data Using SQL*Loader (continued)

- d) Confirm your results by viewing your `lab_17_02_02.log` file in your `/home/oracle/labs` directory.

Practices for Lesson 18

Practice 18-1: Investigating a Critical Error

In this practice, you investigate a critical error. A data block has been corrupted. Use the Enterprise Manager Support Workbench, Health Checkers, and advisors to react to the error.

- 1) Set up the database for the practice. The setup creates a tablespace for the practice, and performs a tablespace backup. Run the `lab_18_01_01.sh` script. The scripts for this practice are in the `$HOME/labs` directory.

```
$ cd $HOME/labs
$ ./lab_18_01_01.sh
```

- 2) Run the `lab_18_01_02.sql` script from SQL*Plus connected as `sysdba` to create the critical error. The script prompts you for a block number; enter the block number given by the script. Notice the error message. In our scenario, the error message could originally be seen by anyone and not reported immediately to the DBA.

Note: Remember to use `oraenv` to set your environment to the `orcl` database if you have not already done so in your terminal window.

```
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.1.0 Production on Wed Jul 15 01:30:42
2009

Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 -
Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options

SQL> @lab_18_01_02.sql
Connected.

FILE_NO    BLOCK_NO
-----  -----
9          129
```

Write down this Block Number because you will need to enter this number when prompted.

```
System altered.

'Enter Block number when prompted'
Enter value for block_no: 129
0+1 records in
0+1 records out
80 bytes (80 B) copied, 6.693e-05 seconds, 1.2 MB/s
0+1 records in
0+1 records out
```

Practice 18-1: Investigating a Critical Error (continued)

```
79 bytes (79 B) copied, 6.3419e-05 seconds, 1.2 MB/s

SELECT * from js.jscopy
*
ERROR at line 1:
ORA-01578: ORACLE data block corrupted (file # 9, block # 129)
ORA-01110: data file 9:
  '/u01/app/oracle/oradata/orcl/jstbs01.dbf'

Disconnected from Oracle Database 11g Enterprise Edition
Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP,
Data Mining
and Real Application Testing options
$
```

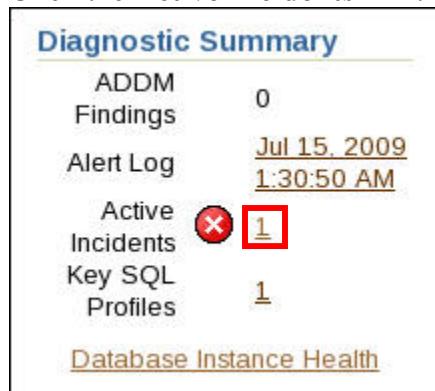
- 3) Log in to Enterprise Manager as the **DBA1** user in the **SYSDBA** role, and then view the alerts on the Database home page and investigate the alert details. When the incident appears in the alerts, click the Active Incidents link.

You should see one or more critical alerts. Depending on the timing, you may see one or more of the following:

Severity	Category	Name	Impact	Message	Alert Triggered
✗	Data Failure	Data Failure Detected		Checker run found 2 new persistent data failures.	Jul 15, 2009 1:32:46 AM
✗	Incident	Oracle Data Block Corruption		An Oracle data block corruption detected in /u01/app/oracle/diag/rdbms/orcl/orcl/alert/log.xml at time/line number: Wed Jul 15 01:30:50 2009/18258.	Jul 15, 2009 1:32:46 AM
⚠	User Audit	Audited User		User SYS logged on from edrsr12p1.us.oracle.com.	Jul 15, 2009 1:17:22 AM

The number of Active Incidents may not match the number of alerts immediately.

Click the **Active Incidents** link.



- 4) Investigate the details of the problem.

On the Support Workbench page and **Problems** tab, expand the problem with the description ORA 1578.

Click the incident number of the most recent (first listed) incident.

Practice 18-1: Investigating a Critical Error (continued)

5) Examine the Incident Details page. Notice that the SQL text of the statement that received the error is listed, and the session information. Depending on the error this information can be very helpful in troubleshooting the error.

6) View a trace file.

Click the **eyeglasses** icon to view the first trace file listed. If you are prompted for Host Login Credentials, enter user `oracle` and password `oracle` and click Continue.

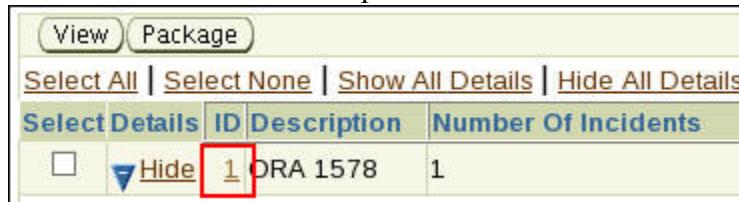
Note: If you do not see a trace file listed, please be patient. It can take a few minutes before the trace file appears in the list.

On the Contents page, you can see the trace file. Sections of the trace file can be expanded or collapsed. The trace file is intended for Oracle internal use. Click **OK**.

Click **Support Workbench** link in the breadcrumb at the top of the page to return to the Support Workbench page.

7) Determine how to resolve the problem.

Click **ID** of the ORA 1578 problem.



Select	Details	ID	Description	Number Of Incidents
<input type="checkbox"/>	Hide	1	ORA 1578	1

8) When the problem details page appears, notice that the Investigate and Resolve section has two tabs that allow you to pursue the solution yourself or enlist the aid of Oracle Support.



Investigate and Resolve

[Go to My Oracle Support](#) [Quick Package](#)

[Self Service](#) [Oracle Support](#)

Assess Damage

[Checker Findings](#)
[Run Checkers](#)
[Database Instance Health](#)

Diagnose

[Alert Log](#)
[Related Problems Across Topology](#)
[Diagnostics for Last Dumped Incident](#)
[Go to My Oracle Support and Research](#)

Resolve

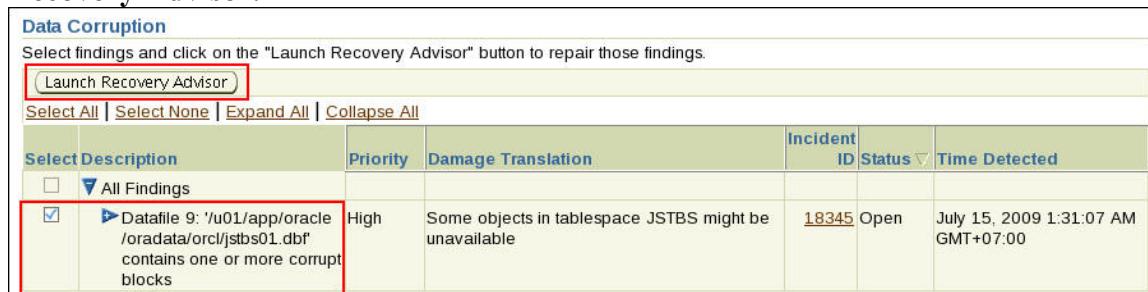
[SQL Repair Advisor](#)
[Data Recovery Advisor](#)

Practice 18-1: Investigating a Critical Error (continued)

9) Get Checker information about the problem. Click **Support Workbench** in the breadcrumb at the top of the page to return to the Support Workbench page. Click **Checker Findings** tab.



10) Get advise. Under the Checker Findings tab, in the Data Corruption section, Select the finding with the description that starts with “Datafile …” and click **Launch Recovery Advisor**.

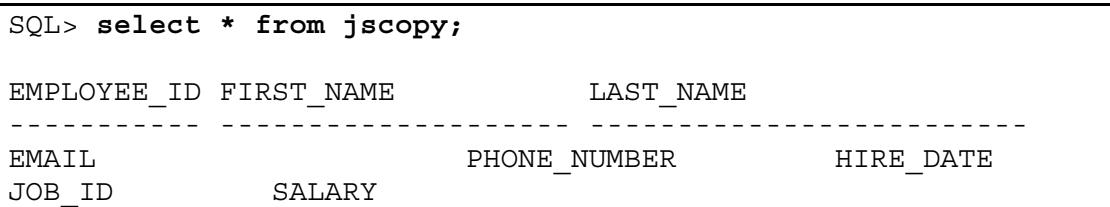


Note: The data file number given in the description is the same as the FILE_NO shown when you ran the `lab_18_01_02.sql` script in step 2.

- On the View and Manage Failures page, confirm that the failure “Datafile …” is selected, and then click Advise.
- The Recovery Advice page recommends an RMAN script. This script will perform a block media recovery. Click Continue.
- The Review page shows the failures that will be resolved. Click Submit Recovery Job.
- The Processing page may appear briefly, after which the Job Activity page appears. The Job Status shows Running. Refresh the page after a few moments and repeat until the page shows No Jobs Found.

11) Open a terminal window and log in to SQL*Plus as the **js** user with a password of **oracle_4U**. Select from the **jscopy** table and ensure that you do not see any further block corruption errors.

Note: Remember to use oraenv to set your environment to the `orcl` database if you have not already done so in your terminal window.



Practice 18-1: Investigating a Critical Error (continued)

```
-----  
COMMISSION_PCT MANAGER_ID DEPARTMENT_ID  
-----  
    100  Steven           King  
SKING                      515.123.4567      17-JUN-87  
AD_PRES          24001  
                  90  
  
    101  Neena           Kochhar  
NKOCHHAR          515.123.4568      21-SEP-89  AD_VP  
17001  
    100           90  
...
```

- When finished, run the `lab_18_01_12.sh` script from the `$HOME/labs` directory to remove the objects created in this practice.

```
$ cd $HOME/labs  
$ ./lab_18_01_12.sh
```

Appendix B

Basic Linux and

vi Commands

***vi* Commands**

The Visual Interpreter/Editor (*vi*) is the most widely used text editor available for the UNIX environment. While almost everybody curses its unwieldy command syntax, it is still the only editor almost certain to be included with every version of the UNIX and Linux operating system. The following are a partial list of available *vi* commands.

vi has two modes. Command line (where anything typed is taken as an editing command) and input mode (where everything typed will be treated as part of the file being edited. To enter the input mode, type a, A, i, I, o, O, c, C, s, S, r, or R. To return to the command-line mode, use the <ESC> key. To access the *vi* editor from SQLPlus, enter the following command:

```
SQL>define _editor=vi
```

To edit a file from SQLPlus prompt, edit <filename> (press enter), from the Linux command prompt, vi <filename> (press enter)

To MOVE the cursor:

h - move left j - move down k - move up l - move right

w - one word forward b - one word backward e - end of current word

W, B, or E - same as lower case but ignores punctuation

0 (zero) - Move to beginning of current line \$ - end of current line

G - go to last line of file H - go to top line on the screen

L - go to last line on screen M - go to middle line on the screen

/<string> - Search forward to the next occurrence of <string>

?<string> - Search backward to the next occurrence of <string>

n - Repeat previous search N - Repeat previous search in opposite direction

<ctrl> f - Scroll forward one page <ctrl> b - Scroll backward one page

To UNDO previous changes:

u - Will undo the most recent change. U - Will undo the most recently deleted text.

:e! - re-edit current file without saving any changes made since last change

To ENTER NEW text:

a - Append text after the current cursor position.

A - Append text to the end of a line (jumps to end of line and begin appending).

c - Change object C - Change from current cursor position to end of the line

i - Insert text before the current cursor position. I - Insert text at the beginning of a line.

o - Insert a blank line BELOW the current cursor position.
O - Insert a blank line ABOVE the current cursor position.
r - Replace character at current cursor position R - Replace all characters until <ESC> is pressed
s - substitute text for character under cursor
:s/A/B/opt substitutes string B for string A. %s/A/B/opt is global replace
options include: g (change all occurrences on current line) c (confirm prior to each change)
p (print changed lines) S - Substitute entire line to the end
. <period> - repeat last change n. <integer><period> repeat last change n times

To leave the input mode, press <ESC>

To DELETE existing text:

x - Will delete the character directly under the current cursor location.
dd - Will delete the entire line where the cursor is located.
dⁿd (where n is some integer) will delete n lines from current cursor position
dw - delete current word D - delete to end of current line
J - Delete return at end of current line. Join this line and the next
<int> J - Join the next <int> lines

COPY, CUT, and PASTE: vi uses a single buffer where the last changed or deleted text is stored. This text may be manipulated with the following commands:

Y - Yank a copy of the current line	y <integer> - Yank a copy of next <int> lines
yw - Yank a copy of the current word	yb - Yank a copy of the previous word
p - Put buffer contents after cursor	P - Put buffer contents before cursor

Also, see the s and S commands under the input section

To SAVE edited changes to an operating system file:

zz - Will terminate edit mode.	:w filename - Will save changes to the filename specified.
:wq - Write all changes and quit the edit mode	

To QUIT without saving changes:

ZZ - Will terminate edit mode.	:q! - Will terminate the file without saving changes.
--------------------------------	-------------------------------------------------------

Basic Linux Commands

This appendix is meant to serve only as a quick reference while you are in class. For more details on these commands, consult the man pages, your Linux documentation, or other Linux command reference books.

Files and Directories	Linux Commands	Description/Comments
Command manual	<code>man <command></code> <code>man -k <string></code> <code>man man</code>	Find the manual entry for this <command>. Show all the manual entries that contain this <string>. Displays the manual page for <code>man</code> .
Command information	<code>info <command></code>	Show the information system entry for this command. Using <code>info info</code> shows a tutorial of the <code>info</code> documentation system.
Print to standard out	<code>cat <file></code>	Concatenate and print – print the named file to the terminal screen.
List users	<code>cat /etc/password</code>	
Change working directory	<code>cd <directory></code>	Change working directory to specified directory cd with no parameters changes to \$HOME.
Copy a file	<code>cp <source_file> <destination_file></code>	Copy a source file to a destination file.
View a file	<code>less <file></code>	View a file a page at a time. This is a GNU version of <code>more</code> , or <code>pg</code> .
View a file	<code>more <file></code>	View a file a page at a time. BSD version.
List directory	<code>ls <directory></code>	Options –l long listing, -R recursive, -a show hidden files, -t sort by time, -r reverse sort, default directory is current working directory.
Create a directory	<code>mkdir <directory></code>	Make a directory defaults into the current working directory, full path may be specified.
Move or rename a file	<code>mv <old_file> <new_file></code>	Move changes the name of a file or moves it to a different directory.

Process List	<code>ps</code> <code>ps -ef</code>	Shows the processes report Shows all processes on the system with a full listing. Many options exist see the man page for details.
Print working directory	<code>pwd</code>	Print to stdout the current working directory.
Remove or erase a file	<code>rm <file></code>	Removing a file on Linux is permanent. Options <code>-r</code> recursive, and <code>-f</code> force (including subdirectories) are <i>very dangerous</i> . Often the <code>rm</code> command is aliased with <code>rm -i</code> . The option <code>-i</code> asks 'Are you sure?'
Create an empty file	<code>touch <file></code>	Create a file.
Name of the machine	<code>hostname</code>	Returns the name of the machine.
The IP address of the machine	<code>host <machine_name></code>	Queries the Domain Name Server, and returns the IP address of the machine name.
Remote shell	<code>rsh <host> <command></code>	Execute a <code><command></code> on <code><host></code> . Rsh is not secure, use ssh instead.
Remote shell	<code>ssh <host></code>	Secure shell, has features to replace rsh, rcp, ftp, and telnet.
Remote shell	<code>telnet <host></code>	Start a terminal session on <code><host></code> . Telnet is not secure use ssh instead.
Search a file for a pattern	<code>grep <option> <pattern> <file></code>	Search a <code><file></code> or stream for a regular expression defined by <code><pattern></code> and show the line that contains that pattern. A common option is <code>-i</code> for case insensitive. grep can accept input from a file or <code>stdin</code> through a pipe as in: <code>netstat -a grep ESTABLISHED</code>
Source a script	<code>. <script_file></code>	In the bash shell this command <code>'.'</code> forces the script to run in the shell. Normal behavior is for the script to run in a child shell.

An interpreter	<code>awk</code>	A macro language for reformatting or interpreting input. For each line of input, a variety of actions can be taken. May be referred to as awk – for “new awk.”
Sort a file	<code>sort</code>	Sort a file takes input from stdin or a filename argument, many options to sort by a particular column, field, etc. See man page.
Command-line editor	<code>sed</code>	Sed is a command-line editor, with many possible commands and options that are very good for editing from a shell script.
Visual editor	<code>vi <file></code>	Terminal based editor available on every Unix system, Linux provides vim, an improved vi, that is a superset of vi.
Gnu editor	<code>emacs <file></code>	This is a GPL editor with extensive customizable features available on most UNIX and Linux distributions.
WSIWIG editor	<code>gedit <file></code>	A full-screen editor, requiring X. Available under Gnome.
WSIWIG	<code>kate <file></code>	A full-screen editor, requires X. Available under KDE
Terminal output	<code>stdout</code>	Standard out (stdout), is not a command but a concept, most Linux commands write to stdout by default unless redirected.
Terminal input (keyboard)	<code>stdin</code>	Standard in (stdin), is not a command but a concept, most Linux commands read from stdin by default unless redirected.
Alias	<code>alias <command> <alias></code>	Make a substitution when a user types <command> substitute and execute <alias>, common alias is alias ‘rm’ ‘rm -i’. These aliases are set in the .bashrc file.
Show shell variables	<code>set</code>	Prints all of the variables that are currently defined in the shell.

Show environment variables	<code>printenv</code> or <code>env</code>	Prints all the environment variables – an environment variable has been ‘exported’ so that it will be inherited by child processes.
File Creation mask	<code>umask -S u=rwx,g=rx,o=rx</code>	Set the default permissions for all files created by this shell or its children. The <code>-S</code> option uses the symbolic notation, the numeric notation is obsolete.
Clock	<code>xclock</code>	An X client that shows a clock on the screen. Often used to test the X windows system.
X access control	<code>xhost</code> <code>xhost +<Xclient></code>	Show the current access control in place. Add a Xclient that is allowed to access the local DISPLAY, if no <code><Xclient></code> is given all are allowed.

System Administration	Linux Commands	Description / Comments
Substitute user	su - username	Change the user that is currently performing the work. This can be used by any user to change who is the effective id of the session user. normal users must provide a password, root does not. The ‘-‘ parameter is optional. It runs the new users login scripts.
Limited root privileges	sudo	The root user may configure which users can execute certain commands as root, and whether a password is required or not. Useful for allowing specific users to perform certain root commands e.g. mount and unmount removable volumes such as CDROMs.
Root file system	/	The root directory for the system directory tree.
Home Directory	/home	Typically the directory in which all user home directories placed. For example: /home/oracle.
Tmp directory	/tmp	A temporary storage area. Do not put anything here you want to keep. SA often have a cron job to remove everything periodically.
Boot directory	/boot	A small partition to hold the kernel image(s) and boot loader instructions.
Log directory	/var/log	The location of most system log files.
Sample configuration files	/etc/inittab	Configuration files are located per the application. Any configuration file that you change after installation should be included in the backup.
Password files	/etc/passwd /etc/shadow	The /etc/passwd file holds user information and must be readable by others; even with encrypted passwords this can be a security hole. The /etc/shadow

		file holds the encrypted passwords and is only readable by root.
Groups file	/etc/group	The /etc/groups file defines the groups on a server and the users that are members of the group; primary group for a user is defined in the /etc/passwd file.
X configuration file	/etc/X11/XF86Config	The file that sets the X server settings for your video card, monitor, mouse, and keyboard. Usually set up with a OS vendor supplied tool.

Schedule a command to run at a regularly scheduled time	crontab -e	Use this command to edit the crontab file, to create the specification for the cron daemon to use.
Schedule a script to run at a particular frequency	/etc/anacrontab	Edit the file to specify a script to run at a particular frequency (see man anacrontab for details).
Schedule a command to run at a single specified time	at <options> TIME	Runs a job specified by <options> at a specified TIME parameter.
Schedule a command	batch <options> <TIME>	Run a command when the load average drops below .8, optionally after a set TIME.
Mount a file system	mount <opt> <dev> <mount_point>	Mount a file system on device <dev> at <mount_point> with the options specified by <dev>.
Unmount a file system	umount <dev> umount <mount_point>	Unmount the file system or device.
Maximum # of user ID	65535	
Recover root password	{lilo} control-x linux S passwd root	This is a procedure to recover the root password if is lost. This requires physical access to the machine and system console. You start by rebooting the machine, then during the LILO boot press and hold [Ctrl] + [x] to get a

	<pre>{grub} c kernel vmlinuz-2.4.9-13 single ro root=/dev/hda8 initrd /initrd-2.4.9-13.img boot passwd root</pre>	<p>prompt and command LILO to boot linux to runlevel S.</p> <p>The second procedure uses the grub boot loader.</p>
Create new user	useradd	<p>The –D option alone shows the defaults.</p> <p>–D with other options changes the defaults options; without –D override, the default (e.g., –g) sets a primary group.</p>

Delete user	userdel	Remove a user and optionally all files belonging to the user.
Modify user account	usermod	Change /etc/password information.
Create new group	groupadd	–g sets the group id; default is first free value above 500.
Delete group	groupdel	Remove a group from the system. May not remove a group that is a primary group for a user. Files owned by deleted group must be manually changed with chown.
Change run levels	init <runlevel>	The init command causes the rcN.d scripts to be evaluated, for the change in run level. init 6 forces a reboot.
Synchronize the disks	sync	Forces the buffer cache and page cache to write all dirty buffers to disk. Used just before a reboot to prevent disk corruption.
Shutdown the Linux system	shutdown <mode> <delay>	Do a graceful shutdown of the system, shut down processes, run all shutdown scripts, and sync disks. The modes are –r, reboot and –h, halt. The delay is a required parameter is a number in seconds or ‘now’. Option shutdown warning message may be sent as well.
Error logs	dmesg	View boot messages. This log is circular, and limited system errors could overwrite boot information after a time.
Network IP configuration	/etc/sysconfig/network-scripts/	This directory holds scripts executed as part of the boot up sequence by rc.sysinit.
Hosts IP addresses	/etc/hosts	A list of hosts that your machine knows about. Must at minimum include the name of the local machine and loopback IP.
Name service switch	/etc/nsswitch.conf	

Network parameters	<code>sysctl -a grep net</code>	View all net parameters that are set for the kernel.
Routing daemon	<code>routed</code>	
NIC Configurations	<code>ifconfig -a</code>	Show all the network devices currently configured.
Secondary IP Address	<code>modprobe ip_alias</code>	
	<code>ifconfig eth0:1 IP</code>	
Login prompt	<code>/etc/issue</code>	Banner message user sees when issued the login prompt.
YP/NIS service binder	<code>/sbin/ypbind</code>	Finds and attaches to a NIS server for name resolution and other services.
Module information	<code>modinfo <options> <module></code>	Display information about kernel modules: <code>-l</code> shows license, <code>-p</code> parameters, <code>-d</code> description.
List modules	<code>lsmod</code>	Show currently loaded modules.
Load module	<code>insmod</code>	Load a loadable module.
Unload module	<code>rmmmod</code>	Unload a loadable module.
Install Software	<code>rpm -ivh package</code>	Install <code>-i</code> , verbose <code>-v</code> , with progress hash marks <code>-h</code> .
Uninstall software	<code>rpm -e package</code>	Erase package <code>-e</code> ; will not uninstall if dependencies exist.
List installed software	<code>rpm -qa</code>	Query <code>-q</code> , All <code>-a</code> , lists all installed packages.
Verify installed software	<code>rpm -V package</code>	Compares installed files with the rpm database information.
List all files	<code>rpm -ql package</code>	List all the files that are part of a package.
Package owner	<code>rpm -qf file</code>	List the package when given the full file name.
Machine model	<code>uname -m</code>	Shows CPU level (e.g., i686).
OS Level	<code>uname -r</code>	Shows kernel version.
Run Level	<code>runlevel</code>	Shows previous and current runlevel.
Kernel Parameters	<code>sysctl -a</code>	Show settings of all settable kernel parameters.
Max # File Descriptors	<code>sysctl fs.file-max</code>	Shows the value of maximum number of file descriptor per process.

Kernel parameter settings	/etc/sysctl.conf	Compiled in kernel parameters; may be reset at bootup by setting them in this file.
Change Kernel Parameter	echo <value> > </proc/<file>	Write the new value of a kernel parameter into the /proc file system.
	echo 2147483648 >/proc/sys/kernel/shmmax	Set the value of the maximum size of a shared memory segment.
Shared Memory	sysctl kernel.shmmax	Show the shmmax parameter.
Change Kernel Parameter	sysctl -w <parameter>=<value>	Change a kernel parameter; the -p option reads the setting from a file and sets them. The default file is /etc/sysctl.conf
Set Process limits	ulimit <option> <value>	Set limits on a shell and processes started by the shell. Users can make limits more restrictive; generally only root can make limit less restrictive; some options require root privilege. Options: -u sets number of processes, -n number of file handles; many others (see man bash).
Show process limits	ulimit	Without options ulimit show the current limit settings.
Interprocess Communication (Shared Memory and Semaphores)	ipcs <option>	Options: -m the current usage of shared memory; -s usage of semaphores; -a shows all.
Remove a shared memory segment	ipcrm shm <shmid>	Releases the shared memory segment identified by <shmid>. <i>This is very dangerous.</i> You can corrupt a database that is using the segment that is released.

System Performance	Linux Commands	Description / Comments
Performance monitor	top	View real-time OS and process statistics.
System activity reporter	sar --<options> <interval> <count>	Options: -q shows CPU queue, -u CPU utilization, -d device activity, -n DEV network device activity, many more (see man page). Interval is in seconds.
Virtual Memory statistics	vmstat <interval> < count>	Interval is in seconds.
Virtual Memory statistics	cat /proc/meminfo	Shows instantaneous virtual memory usage.
Kernel Cache statistics	cat /proc/slabinfo	Kernel slab allocator statistics: frequently allocated cache objects such as inode, dentries, and asynchronous IO buffers.
I/O statistics	iostat <option> <interval> <count>	Options: -d device activity, -c CPU activity, -x extended disk activity statistics. The interval is in seconds.
Multiprocessor Statistics	mpstat -P <cpu> <count> <interval>	Return CPU statistics for particular processor or <i>all</i> CPUs in an smp system.
Physical RAM	64 GB (Theoretical)	Maximum physical RAM requires enterprise kernel (Red Hat Enterprise Linux AS 21 only supports up to 16 GB).
Swap device	swapon -s	Shows devices currently in use for swap. The swap device is arbitrary designated at install. It may be changed or added to. Multiple swap devices may be created; swap size should be at least as large as physical memory.

Display swap size	<code>free</code>	Show the current memory and swap usage.
Activate Swap	<code>swapon -a</code>	Turn on swap.
Free disk blocks	<code>df -k</code>	Measured in KB; use <code>-m</code> for MB units.
Device listing	<code>cat /proc/devices</code>	List devices known to the system by major and minor number.
Disk information	<code>cat /proc/scsi/scsi0/sda/model</code> <code>cat /proc/ide/ide0/hda/model</code>	View SCSI disk information. View IDE disk information.
Print network statistics	<code>netstat <options></code>	Print a wide variety of network statistics (see <code>man netstat</code>).
Graphical system statistics viewer	<code>xosview</code>	An X-based display of recent OS statistics.

Misc System Information	Linux Commands	Description / Comments
NFS exported	/etc/exports	Database file are not supported on simple NFS.
NFS Client mounted directories	/var/lib/nfs/xtab	
Max File System	2 TB with 4KB block size (on 32 kernel)	With ext3 and ext2, others vary.
Max File Size	2 GB {512B block size}	The oracle database can create files up to 64 GB with a 16 KB database block size.
File size can not exceed file system	2 TB {4KB block size}	The 32-bit kernel limits file and block devices to 2 TB.
File System Block size	dump2fs <device>	Dump the file system properties to stdout.
Filesystem table	/etc/fstab	Mounts these file systems at boot up.
Journal Filesystem types	ext3 reiserfs	
Disk Label	fdisk -l	fdisk is not available on all distributions.
Extend File system	resize2fs resize_reiserfs	Extending a file system is applicable to only some file system types.
Backup	tar cvf /dev/rst0 /	Create a backup of the root / file system.
Restore	tar xvf /dev/rst0	Restore the root / file system.
Prepare boot volumes	/sbin/lilo	Must be run after changing /etc/lilo.conf to push changes to boot loader.
Startup script	/etc/rc.d/rc	
Kernel	/boot/vmlinuz	
Kernel Bits	getconf WORD_BIT	POSIX call to get kernel information. There are many other variables besides WORD_BIT.

Boot single user	<pre>{lilo} control-x linux S {grub} c kernel vmlinuz-2.4.9-13 single ro root=/dev/hda8 initrd /initrd-2.4.9-13.img boot</pre>	Use LILO facility. Use GRUB Boot Loader.
Time zone Management	/etc/sysconfig/clock	
SW Directory	/var/lib/rpm	Directory where rpm database are kept.
Devices	/dev	This directory holds all the device files.
CPU	cat /proc/cpuinfo	Shows CPU static information.
Whole Disk	/dev/sda	Device name.
CDROM	/dev/cdrom	Usually mounted at /mnt/cdrom.
CDROM file type	iso9660	
Floppy drive	/dev/fd0	Usually mounted at /mnt/floppy.
System information	/proc	The /proc filesystem is a memory-based file system that allows access to process and kernel settings and statistics.
Compile and link a executable	make -f <file> <command>	Use a make file <file> to determine which parts of a large program need to be recompiled, and issue the commands required to compile, link, and prepare the executable for use.

LVM	Linux (UnitedLinux)	Description / Comments
LVM	Logical Volume Manager	This package is not provided by Red Hat Enterprise Linux AS 2.1 and may not be added without tainting the kernel. Kernel support is provided in United Linux.
LVM Concepts	logical extents	A Logical volume is made up of logical extents.
	logical volume	A set of logical extents taken from a volume group and presented to the OS as a disk volume. These extents may be striped across multiple disks.
	volume group	A set of physical disk partitions created by fdisk or the like, initialized with pvcreate, then grouped into a physical volume with vgcreate.
Display volume group	vgdisplay -v	
Modify physical volume	pvchange	
Prepare physical disk	pvcreate	
List physical volume	pvdisplay	
Remove disk from volume group	vgreduce	
Move logical volumes to another physical volumes	pvmove	
Create volume group	vgcreate	
Remove volume group	vgremove	
Volume group availability	vgchange	
Restore volume group	vgcfgrestore	

Exports volume group	<code>vgexport</code>	
Imports volume group	<code>vgimport</code>	
Volume group listing	<code>vgscan</code>	
Change logical volume characteristics	<code>lvchange</code>	
List logical volume	<code>lvdisplay</code>	
Make logical volume	<code>lvcreate</code>	
Extend logical volume	<code>lvextend</code>	
Reduce logical volume	<code>lvreduce</code>	
Remove logical volume	<code>lvremove</code>	
Create striped volumes	<code>lvcreate -i 3 -I 64</code>	

Appendix C

SQL Statement Syntax

SQL Statements

This excerpt from the Oracle Database SQL Quick Reference guide presents the syntax for Oracle SQL statements. SQL statements are the means by which programs and users access data in an Oracle database.

Table 1 shows each SQL statement and its related syntax. Table 2 shows the syntax of the subclauses found in the table 1.

See Also: Oracle Database SQL Reference for detailed information about Oracle SQL

Table 1: Syntax for SQL Statements

SQL Statement	Syntax
ALTER CLUSTER	<pre>ALTER CLUSTER [schema.]cluster { physical_attributes_clause SIZE size_clause allocate_extent_clause deallocate_unused_clause { CACHE NOCACHE } } [physical_attributes_clause SIZE size_clause allocate_extent_clause deallocate_unused_clause { CACHE NOCACHE }]... [parallel_clause] ;</pre>
ALTER DATABASE	<pre>ALTER DATABASE [database] { startup_clauses recovery_clauses database_file_clauses logfile_clauses controlfile_clauses standby_database_clauses default_settings_clauses redo_thread_clauses security_clause } ;</pre>
ALTER DIMENSION	<pre>ALTER DIMENSION [schema.]dimension { ADD { level_clause hierarchy_clause attribute_clause extended_attribute_clause } [ADD</pre>

SQL Statement	Syntax
	<pre> { level_clause hierarchy_clause attribute_clause extended_attribute_clause }] ... DROP { LEVEL level [RESTRICT CASCADE] HIERARCHY hierarchy ATTRIBUTE attribute [LEVEL level [COLUMN column , COLUMN column]...] } [DROP { LEVEL level [RESTRICT CASCADE] HIERARCHY hierarchy ATTRIBUTE attribute [LEVEL level [COLUMN column , COLUMN column]...] }]] ... COMPILE } ; </pre>
ALTER DISKGROUP	<pre> ALTER DISKGROUP { disk_clauses diskgroup_clauses } [{ disk_clauses diskgroup_clauses }]... ; </pre>
ALTER FUNCTION	<pre> ALTER FUNCTION [schema.]function COMPILE [DEBUG] [compiler_parameters_clause [compiler_parameters_clause] ...] [REUSE SETTINGS] ; </pre>
ALTER INDEX	<pre> ALTER INDEX [schema.]index { { deallocate_unused_clause allocate_extent_clause shrink_clause parallel_clause physical_attributes_clause logging_clause } [deallocate_unused_clause allocate_extent_clause shrink_clause parallel_clause physical_attributes_clause logging_clause]... rebuild_clause PARAMETERS ('ODCI_parameters') </pre>

SQL Statement	Syntax
	<pre> { ENABLE DISABLE } UNUSABLE RENAME TO new_name COALESCE { MONITORING NOMONITORING } USAGE UPDATE BLOCK REFERENCES alter_index_partitioning } ; </pre>
ALTER INDEXTYPE	<pre> ALTER INDEXTYPE [schema.] indextype { { ADD DROP } [schema.] operator (parameter_types) [, { ADD DROP } [schema.] operator (parameter_types)]... [using_type_clause] COMPILE } ; </pre>
ALTER JAVA	<pre> ALTER JAVA { SOURCE CLASS } [schema.] object_name [RESOLVER ((match_string [,] { schema_name - }) [(match_string [,] { schema_name - })]...)] { { COMPILE RESOLVE } invoker_rights_clause } ; </pre>
ALTER MATERIALIZED VIEW	<pre> ALTER MATERIALIZED VIEW [schema.] (materialized_view) [physical_attributes_clause table_compression LOB_storage_clause [, LOB_storage_clause]... modify_LOB_storage_clause [, modify_LOB_storage_clause]... alter_table_partitioning parallel_clause logging_clause allocate_extent_clause shrink_clause { CACHE NOCACHE }] [alter_iot_clauses] [USING INDEX physical_attributes_clause] [MODIFY scoped_table_ref_constraint alter_mv_refresh] [{ ENABLE DISABLE } QUERY REWRITE COMPILE] </pre>

SQL Statement	Syntax
	<pre> CONSIDER FRESH] ; </pre>
ALTER MATERIALIZED VIEW LOG	<pre> ALTER MATERIALIZED VIEW LOG [FORCE] ON [schema.]table [physical_attributes_clause alter_table_partitioning parallel_clause logging_clause allocate_extent_clause shrink_clause { CACHE NOCACHE }] [ADD { { OBJECT ID PRIMARY KEY ROWID SEQUENCE } [(column [, column]...)] (column [, column]...) } [, { { OBJECT ID PRIMARY KEY ROWID SEQUENCE } [(column [, column]...)] (column [, column]...) }] ... [new_values_clause]] ; </pre>
ALTER OPERATOR	<pre> ALTER OPERATOR [schema.]operator { add_binding_clause drop_binding_clause COMPILE } ; </pre>
ALTER OUTLINE	<pre> ALTER OUTLINE [PUBLIC PRIVATE] outline { REBUILD RENAME TO new_outline_name CHANGE CATEGORY TO new_category_name { ENABLE DISABLE } } [REBUILD RENAME TO new_outline_name CHANGE CATEGORY TO new_category_name { ENABLE DISABLE }]... ; </pre>

SQL Statement	Syntax
ALTER PACKAGE	<pre>ALTER PACKAGE [schema.]package COMPILE [DEBUG] [PACKAGE SPECIFICATION BODY] [compiler_parameters_clause [compiler_parameters_clause] ...] [REUSE SETTINGS] ;</pre>
ALTER PROCEDURE	<pre>ALTER PROCEDURE [schema.]procedure COMPILE [DEBUG] [compiler_parameters_clause [compiler_parameters_clause] ...] [REUSE SETTINGS] ;</pre>
ALTER PROFILE	<pre>ALTER PROFILE profile LIMIT { resource_parameters password_parameters } [resource_parameters password_parameters]... ;</pre>
ALTER RESOURCE COST	<pre>ALTER RESOURCE COST { CPU_PER_SESSION CONNECT_TIME LOGICAL_READS_PER_SESSION PRIVATE_SGA } integer [{ CPU_PER_SESSION CONNECT_TIME LOGICAL_READS_PER_SESSION PRIVATE_SGA } integer] ... ;</pre>
ALTER ROLE	<pre>ALTER ROLE role { NOT IDENTIFIED IDENTIFIED { BY password USING [schema.]package EXTERNALLY GLOBALLY } } ;</pre>
ALTER ROLEBACK SEGMENT	<pre>ALTER ROLLBACK SEGMENT rollback_segment { ONLINE OFFLINE storage_clause SHRINK [TO integer [K M]] };</pre>

SQL Statement	Syntax
ALTER SEQUENCE	<pre> ALTER SEQUENCE [schema.]sequence { INCREMENT BY integer { MAXVALUE integer NOMAXVALUE } { MINVALUE integer NOMINVALUE } { CYCLE NOCYCLE } { CACHE integer NOCACHE } { ORDER NOORDER } } [INCREMENT BY integer { MAXVALUE integer NOMAXVALUE } { MINVALUE integer NOMINVALUE } { CYCLE NOCYCLE } { CACHE integer NOCACHE } { ORDER NOORDER }]... ; </pre>
ALTER SESSION	<pre> ALTER SESSION { ADVISE { COMMIT ROLLBACK NOTHING } CLOSE DATABASE LINK dblink { ENABLE DISABLE } COMMIT IN PROCEDURE { ENABLE DISABLE } GUARD { ENABLE DISABLE FORCE } PARALLEL { DML DDL QUERY } [PARALLEL integer] { ENABLE RESUMABLE [TIMEOUT integer] [NAME string] DISABLE RESUMABLE } alter_session_set_clause } ; </pre>
ALTER SYSTEM	<pre> ALTER SYSTEM { archive_log_clause checkpoint_clause check_datafiles_clause DUMP ACTIVE SESSION HISTORY [MINUTES integer] distributed_recov_clauses restricted_session_clauses FLUSH { SHARED_POOL BUFFER_CACHE } end_session_clauses SWITCH LOGFILE { SUSPEND RESUME } quiesce_clauses shutdown_dispatcher_clause REGISTER SET alter_system_set_clause [alter_system_set_clause]... RESET alter_system_reset_clause [alter_system_reset_clause]... } ; </pre>
ALTER TABLE	<pre> ALTER TABLE [schema.]table [alter_table_properties column_clauses] </pre>

SQL Statement	Syntax
	<pre> constraint_clauses alter_table_partitioning alter_external_table_clauses move_table_clause] [enable_disable_clause { ENABLE DISABLE } { TABLE LOCK ALL TRIGGERS } [enable_disable_clause { ENABLE DISABLE } { TABLE LOCK ALL TRIGGERS }] ...]; </pre>
ALTER TABLESPACE	<pre> ALTER TABLESPACE tablespace { DEFAULT [table_compression] storage_clause MINIMUM EXTENT integer [K M] RESIZE size_clause COALESCE RENAME TO new_tablespace_name { BEGIN END } BACKUP datafile_tempfile_clauses tablespace_logging_clauses tablespace_group_clause tablespace_state_clauses autoextend_clause flashback_mode_clause tablespace_retention_clause }; </pre>
ALTER TRIGGER	<pre> ALTER TRIGGER [schema.]trigger { ENABLE DISABLE RENAME TO new_name COMPILE [DEBUG] [compiler_parameters_clause [compiler_parameters_clause] ...] [REUSE SETTINGS] }; </pre>
ALTER TYPE	<pre> ALTER TYPE [schema.]type { compile_type_clause replace_type_clause { alter_method_spec alter_attribute_definition alter_collection_clauses [NOT] { INSTANTIABLE FINAL } } [dependent_handling_clause] }; </pre>

SQL Statement	Syntax
ALTER USER	<pre> ALTER USER { user { IDENTIFIED { BY password [REPLACE old_password] EXTERNALLY GLOBALLY AS 'external_name' } DEFAULT TABLESPACE tablespace TEMPORARY TABLESPACE { tablespace tablespace_group_name } QUOTA { integer [K M] UNLIMITED } ON tablespace [QUOTA { integer [K M] UNLIMITED } ON tablespace]... PROFILE profile DEFAULT ROLE { role [, role]... ALL [EXCEPT role [, role]... NONE } PASSWORD EXPIRE ACCOUNT { LOCK UNLOCK } } [{ IDENTIFIED { BY password [REPLACE old_password] EXTERNALLY GLOBALLY AS 'external_name' } DEFAULT TABLESPACE tablespace TEMPORARY TABLESPACE { tablespace tablespace_group_name } QUOTA { integer [K M] UNLIMITED } ON tablespace [QUOTA { integer [K M] UNLIMITED } ON tablespace]... PROFILE profile DEFAULT ROLE { role [, role]... ALL [EXCEPT role [, role]... NONE } PASSWORD EXPIRE ACCOUNT { LOCK UNLOCK } }]... user [, user]... proxy_clause ; </pre>

SQL Statement	Syntax
ALTER VIEW	<pre>ALTER VIEW [schema.]view { ADD out_of_line_constraint MODIFY CONSTRAINT constraint { RELY NORELY } DROP { CONSTRAINT constraint PRIMARY KEY UNIQUE (column [, column]...) } COMPILE } ;</pre>
ANALYZE	<pre>ANALYZE { TABLE [schema.]table [PARTITION (partition) SUBPARTITION (subpartition)] INDEX [schema.]index [PARTITION (partition) SUBPARTITION (subpartition)] CLUSTER [schema.]cluster } { validation_clauses LIST CHAINED ROWS [into_clause] DELETE [SYSTEM] STATISTICS compute_statistics_clause estimate_statistics_clause } ;</pre>
ASSOCIATE STATISTICS	<pre>ASSOCIATE STATISTICS WITH { column_association function_association } ;</pre>
AUDIT	<pre>AUDIT { sql_statement_clause schema_object_clause } [BY { SESSION ACCESS }] [WHENEVER [NOT] SUCCESSFUL] ;</pre>
CALL	<pre>CALL { routine_clause object_access_expression } [INTO :host_variable [[INDICATOR] :indicator_variable]] ;</pre>
COMMENT	<pre>COMMENT ON { TABLE [schema.] { table view } COLUMN [schema.] { table. view. materialized_view. } column OPERATOR [schema.] operator INDEXTYPE [schema.] indextype</pre>

SQL Statement	Syntax
	<pre> MATERIALIZED VIEW materialized_view IS 'text' ; </pre>
COMMIT	<pre> COMMIT [WORK] [COMMENT 'text' FORCE 'text' [, integer]] ; </pre>
CREATE CLUSTER	<pre> CREATE CLUSTER [schema.]cluster (column datatype [SORT] [, column datatype [SORT]]...) [{ physical_attributes_clause SIZE size_clause TABLESPACE tablespace { INDEX [SINGLE TABLE] HASHKEYS integer [HASH IS expr] } } [physical_attributes_clause SIZE size_clause TABLESPACE tablespace { INDEX [SINGLE TABLE] HASHKEYS integer [HASH IS expr] }]]... [parallel_clause] [NOROWDEPENDENCIES ROWDEPENDENCIES] [CACHE NOCACHE] ; </pre>
CREATE CONTEXT	<pre> CREATE [OR REPLACE] CONTEXT namespace USING [schema.] package [INITIALIZED { EXTERNALLY GLOBALLY } ACCESSED GLOBALLY] ; </pre>
CREATE CONTROLFILE	<pre> CREATE CONTROLFILE [REUSE] [SET] DATABASE database [logfile_clause] { RESETLOGS NORESETLOGS } [DATAFILE file_specification [, file_specification]...] [{ MAXLOGFILES integer MAXLOGMEMBERS integer MAXLOGHISTORY integer </pre>

SQL Statement	Syntax
	<pre> MAXDATAFILES integer MAXINSTANCES integer { ARCHIVELOG NOARCHIVELOG } FORCE LOGGING } [MAXLOGFILES integer MAXLOGMEMBERS integer MAXLOGHISTORY integer MAXDATAFILES integer MAXINSTANCES integer { ARCHIVELOG NOARCHIVELOG } FORCE LOGGING]... [character_set_clause] ; </pre>
CREATE DATABASE	<pre> CREATE DATABASE [database] { USER SYS IDENTIFIED BY password USER SYSTEM IDENTIFIED BY password CONTROLFILE REUSE MAXDATAFILES integer MAXINSTANCES integer CHARACTER SET charset NATIONAL CHARACTER SET charset SET DEFAULT { BIGFILE SMALLFILE } TABLESPACE database_logging_clauses tablespace_clauses set_time_zone_clause }... ; </pre>
CREATE DATABASE LINK	<pre> CREATE [SHARED] [PUBLIC] DATABASE LINK dblink [CONNECT TO { CURRENT_USER user IDENTIFIED BY password [dblink_authentication] } dblink_authentication] [USING 'connect_string'] ; </pre>
CREATE DIMENSION	<pre> CREATE DIMENSION [schema.]dimension level_clause [level_clause]... { hierarchy_clause attribute_clause extended_attribute_clause } [hierarchy_clause attribute_clause extended_attribute_clause]... ; </pre>

SQL Statement	Syntax
CREATE DIRECTORY	CREATE [OR REPLACE] DIRECTORY directory AS 'path_name' ;
CREATE DISKGROUP	CREATE DISKGROUP diskgroup_name [{ HIGH NORMAL EXTERNAL } REDUNDANCY] [FAILGROUP failgroup_name] DISK qualified_disk_clause [, qualified_disk_clause]... [[FAILGROUP failgroup_name] DISK qualified_disk_clause [, qualified_disk_clause]...]... ;
CREATE FUNCTION	CREATE [OR REPLACE] FUNCTION [schema.]function [(argument [IN OUT IN OUT] [NOCOPY] datatype [, argument [IN OUT IN OUT] [NOCOPY] datatype]...)] RETURN datatype [{ invoker_rights_clause DETERMINISTIC parallel_enable_clause } [invoker_rights_clause DETERMINISTIC parallel_enable_clause]...] { { AGGREGATE PIPELINED } USING [schema.]implementation_type [PIPELINED] { IS AS } { pl/sql_function_body call_spec } } ;
CREATE INDEX	CREATE [UNIQUE BITMAP] INDEX [schema.]index ON { cluster_index_clause table_index_clause bitmap_join_index_clause } ;
CREATE INDEXTYPE	CREATE [OR REPLACE] INDEXTYPE [schema.]inextype FOR [schema.]operator (paramater_type [, paramater_type]...) [, [schema.]operator (paramater_type [, paramater_type]...)]... using_type_clause ;

SQL Statement	Syntax
CREATE JAVA	<pre> CREATE [OR REPLACE] [AND { RESOLVE COMPILE }] [NOFORCE] JAVA { { SOURCE RESOURCE } NAMED [schema.]primary_name CLASS [SCHEMA schema] } [invoker_rights_clause] [RESOLVER ((match_string [,] { schema_name - }) [(match_string [,] { schema_name - })]...)] { USING { BFILE (directory_object_name , server_file_name) { CLOB BLOB BFILE } subquery 'key_for_BLOB' } AS source_text } ; </pre>
CREATE LIBRARY	<pre> CREATE [OR REPLACE] LIBRARY [schema.]libname { IS AS } 'filename' [AGENT 'agent_dblink'] ; </pre>
CREATE MATERIALIZED VIEW	<pre> CREATE MATERIALIZED VIEW [schema.]materialized_view [OF [schema.]object_type] [(scoped_table_ref_constraint)] { ON PREBUILT TABLE [{ WITH WITHOUT } REDUCED PRECISION] physical_properties materialized_view_props } [USING INDEX [physical_attributes_clause TABLESPACE tablespace] [physical_attributes_clause TABLESPACE tablespace]... USING NO INDEX] [create_mv_refresh] [FOR UPDATE] [{ DISABLE ENABLE } QUERY REWRITE] AS subquery ; </pre>
CREATE MATERIALIZED VIEW LOG	<pre> CREATE MATERIALIZED VIEW LOG ON [schema.] table [physical_attributes_clause TABLESPACE tablespace] </pre>

SQL Statement	Syntax
	<pre> logging_clause { CACHE NOCACHE } physical_attributes_clause TABLESPACE tablespace logging_clause { CACHE NOCACHE }]...] [parallel_clause] [table_partitioning_clauses] [WITH { OBJECT ID PRIMARY KEY ROWID SEQUENCE (column [, column]...) } [, { OBJECT ID PRIMARY KEY ROWID SEQUENCE (column [, column]...) }]]... [new_values_clause]] ; </pre>
CREATE OPERATOR	<pre>CREATE [OR REPLACE] OPERATOR [schema.] operator binding_clause ;</pre>
CREATE OUTLINE	<pre>CREATE [OR REPLACE] [PUBLIC PRIVATE] OUTLINE [outline] [FROM [PUBLIC PRIVATE] source_outline] [FOR CATEGORY category] [ON statement] ;</pre>
CREATE PACKAGE	<pre>CREATE [OR REPLACE] PACKAGE [schema.]package [invoker_rights_clause] { IS AS } pl/sql_package_spec ;</pre>
CREATE PACKAGE BODY	<pre>CREATE [OR REPLACE] PACKAGE BODY [schema.]package { IS AS } pl/sql_package_body ;</pre>
CREATE PFILE	<pre>CREATE PFILE [= 'pfile_name'] FROM SPFILE [= 'spfile_name'] ;</pre>
CREATE PROCEDURE	<pre>CREATE [OR REPLACE] PROCEDURE [schema.]procedure [(argument [IN OUT IN OUT] [NOCOPY] datatype [, argument [IN OUT IN OUT] [NOCOPY]</pre>

SQL Statement	Syntax
	<pre> datatype] ...)] [invoker_rights_clause] { IS AS } { pl/sql_subprogram_body call_spec } ; </pre>
CREATE PROFILE	<pre> CREATE PROFILE profile LIMIT { resource_parameters password_parameters } [resource_parameters password_parameters] ... ; </pre>
CREATE ROLE	<pre> CREATE ROLE role [NOT IDENTIFIED IDENTIFIED { BY password USING [schema.] package EXTERNALLY GLOBALLY }]] ; </pre>
CREATE ROLEBACK SEGMENT	<pre> CREATE [PUBLIC] ROLLBACK SEGMENT rollback_segment [{ TABLESPACE tablespace storage_clause } [TABLESPACE tablespace storage_clause]...]; </pre>
CREATE SCHEMA	<pre> CREATE SCHEMA AUTHORIZATION schema { create_table_statement create_view_statement grant_statement } [create_table_statement create_view_statement grant_statement]... ; </pre>
CREATE SEQUENCE	<pre> CREATE SEQUENCE [schema.] sequence [{ INCREMENT BY START WITH } integer { MAXVALUE integer NOMAXVALUE } { MINVALUE integer NOMINVALUE } { CYCLE NOCYCLE } { CACHE integer NOCACHE } { ORDER NOORDER }] [{ INCREMENT BY START WITH } integer { MAXVALUE integer NOMAXVALUE } { MINVALUE integer NOMINVALUE } { CYCLE NOCYCLE }] </pre>

SQL Statement	Syntax
	<pre> { CACHE integer NOCACHE } { ORDER NOORDER }]... ; </pre>
CREATE SPFILE	CREATE SPFILE [= 'spfile_name'] FROM PFILE [= 'pfile_name'] ;
CREATE SYNONYM	CREATE [OR REPLACE] [PUBLIC] SYNONYM [schema.]synonym FOR [schema.]object [@ dblink] ;
CREATE TABLE	{ relational_table object_table XMLType_table }
CREATE TABLESPACE	CREATE [BIGFILE SMALLFILE] { permanent_tablespace_clause temporary_tablespace_clause undo_tablespace_clause } ;
CREATE TRIGGER	CREATE [OR REPLACE] TRIGGER [schema.]trigger { BEFORE AFTER INSTEAD OF } { dml_event_clause { ddl_event [OR ddl_event]... database_event [OR database_event]... } ON { [schema.]SCHEMA DATABASE } } [WHEN (condition) { pl/sql_block call_procedure_statement } ;
CREATE TYPE	{ create_incomplete_type create_object_type create_varray_type create_nested_table_type }
CREATE TYPE BODY	CREATE [OR REPLACE] TYPE BODY [schema.]type_name { IS AS } { subprogram_declaration map_order_func_declaration } [; { subprogram_declaration map_order_func_declaration }]]... END ;

SQL Statement	Syntax
CREATE USER	<pre> CREATE USER user IDENTIFIED { BY password EXTERNALLY GLOBALLY AS 'external_name' } [DEFAULT TABLESPACE tablespace TEMPORARY TABLESPACE { tablespace tablespace_group_name } QUOTA { integer [K M] UNLIMITED } ON tablespace [QUOTA { integer [K M] UNLIMITED } ON tablespace]... PROFILE profile PASSWORD EXPIRE ACCOUNT { LOCK UNLOCK } [DEFAULT TABLESPACE tablespace TEMPORARY TABLESPACE { tablespace tablespace_group_name } QUOTA { integer [K M] UNLIMITED } ON tablespace [QUOTA { integer [K M] UNLIMITED } ON tablespace]... PROFILE profile PASSWORD EXPIRE ACCOUNT { LOCK UNLOCK }]...] ; </pre>
CREATE VIEW	<pre> CREATE [OR REPLACE] [[NO] FORCE] VIEW [schema.]view [(alias [inline_constraint [inline_constraint]... out_of_line_constraint [, alias [inline_constraint [inline_constraint]... out_of_line_constraint]...) object_view_clause XMLType_view_clause] AS subquery [subquery_restriction_clause] ; </pre>

SQL Statement	Syntax
DELETE	<pre>DELETE [hint] [FROM] { dml_table_expression_clause ONLY (dml_table_expression_clause) } [t_alias] [where_clause] [returning_clause] ;</pre>
DISASSOCIATE STATISTICS	<pre>DISASSOCIATE STATISTICS FROM { COLUMNS [schema.]table.column [, [schema.]table.column]... FUNCTIONS [schema.]function [, [schema.]function]... PACKAGES [schema.]package [, [schema.]package]... TYPES [schema.]type [, [schema.]type]... INDEXES [schema.]index [, [schema.]index]... INDEXTYPES [schema.]indextype [, [schema.]indextype]... } [FORCE] ;</pre>
DROP CLUSTER	<pre>DROP CLUSTER [schema.]cluster [INCLUDING TABLES [CASCADE CONSTRAINTS]] ;</pre>
DROP CONTEXT	<pre>DROP CONTEXT namespace ;</pre>
DROP DATABASE	<pre>DROP DATABASE ;</pre>
DROP DATABASE LINK	<pre>DROP [PUBLIC] DATABASE LINK dblink ;</pre>
DROP DIMENSION	<pre>DROP DIMENSION [schema.]dimension ;</pre>
DROP DIRECTORY	<pre>DROP DIRECTORY directory_name ;</pre>
DROP DISKGROUP	<pre>DROP DISKGROUP diskgroup_name [{ INCLUDING EXCLUDING } CONTENTS] ;</pre>
DROP FUNCTION	<pre>DROP FUNCTION [schema.]function_name ;</pre>
DROP INDEX	<pre>DROP INDEX [schema.]index [FORCE] ;</pre>

SQL Statement	Syntax
DROP INDEXTYPE	DROP INDEXTYPE [schema.]indextype [FORCE] ;
DROP JAVA	DROP JAVA { SOURCE CLASS RESOURCE } [schema.]object_name ;
DROP LIBRARY	DROP LIBRARY library_name ;
DROP MATERIALIZED VIEW	DROP MATERIALIZED VIEW [schema.]materialized_view [PRESERVE TABLE] ;
DROP MATERIALIZED VIEW LOG	DROP MATERIALIZED VIEW LOG ON [schema.]table ;
DROP OPERATOR	DROP OPERATOR [schema.]operator [FORCE] ;
DROP OUTLINE	DROP OUTLINE outline ;
DROP PACKAGE	DROP PACKAGE [BODY] [schema.]package ;
DROP PROCEDURE	DROP PROCEDURE [schema.]procedure ;
DROP PROFILE	DROP PROFILE profile [CASCADE] ;
DROP ROLE	DROP ROLE role ;
DROP ROLLBACK SEGMENT	DROP ROLLBACK SEGMENT rollback_segment ;
DROP SEQUENCE	DROP SEQUENCE [schema.]sequence_name ;
DROP SYNONYM	DROP [PUBLIC] SYNONYM [schema.]synonym [FORCE] ;
DROP TABLE	DROP TABLE [schema.]table [CASCADE CONSTRAINTS] [PURGE] ;
DROP TABLESPACE	DROP TABLESPACE tablespace [INCLUDING CONTENTS [AND DATAFILES] [CASCADE CONSTRAINTS]] ;
DROP TRIGGER	DROP TRIGGER [schema.]trigger ;

SQL Statement	Syntax
DROP TYPE	DROP TYPE [schema.]type_name [FORCE VALIDATE] ;
DROP TYPE BODY	DROP TYPE BODY [schema.]type_name ;
DROP USER	DROP USER user [CASCADE] ;
DROP VIEW	DROP VIEW [schema.] view [CASCADE CONSTRAINTS] ;
EXPLAIN PLAN	EXPLAIN PLAN [SET STATEMENT_ID = 'text'] [INTO [schema.]table [@ dblink]] FOR statement ;
FLASHBACK DATABASE	FLASHBACK [STANDBY] DATABASE [database] { TO { SCN TIMESTAMP } expr TO BEFORE { SCN TIMESTAMP } expr } ;
FLASHBACK TABLE	FLASHBACK TABLE [schema.]table [, [schema.]table]... TO { { SCN TIMESTAMP } expr { ENABLE DISABLE } TRIGGERS BEFORE DROP [RENAME TO table] } ;
GRANT	GRANT { grant_system_privileges grant_object_privileges } ;
INSERT	INSERT [hint] { single_table_insert multi_table_insert } ;
LOCK TABLE	LOCK TABLE [schema.] { table view } [{ PARTITION (partition) SUBPARTITION (subpartition) } @ dblink] [, [schema.] { table view } [{ PARTITION (partition) SUBPARTITION (subpartition) } @ dblink]

SQL Statement	Syntax
	<pre data-bbox="698 255 948 354">]... IN lockmode MODE [NOWAIT] ;</pre>
MERGE	<pre data-bbox="638 397 1405 599"> MERGE [hint] INTO [schema.]table [t_alias] USING [schema.] { table view subquery } [t_alias] ON (condition) [merge_update_clause] [merge_insert_clause] ;</pre>
NOAUDIT	<pre data-bbox="638 650 1209 853"> NOAUDIT { sql_statement_clause [, sql_statement_clause]... schema_object_clause [, schema_object_clause]... } [WHENEVER [NOT] SUCCESSFUL] ;</pre>
PURGE	<pre data-bbox="638 903 1217 1127"> PURGE { { TABLE table INDEX index } { RECYCLEBIN DBA_RECYCLEBIN } TABLESPACE tablespace [USER user] } ;</pre>
RENAME	<pre data-bbox="638 1170 894 1227"> RENAME old_name TO new_name ;</pre>
REVOKE	<pre data-bbox="638 1277 1171 1355"> REVOKE { revoke_system_privileges revoke_object_privileges } ;</pre>
ROLLBACK	<pre data-bbox="638 1404 1135 1522"> ROLLBACK [WORK] [TO [SAVEPOINT] savepoint FORCE 'text'] ;</pre>
SAVEPOINT	<pre data-bbox="638 1571 972 1598"> SAVEPOINT savepoint ;</pre>
SELECT	<pre data-bbox="638 1636 1148 1664"> subquery [for_update_clause] ;</pre>
SET CONSTRAINT[S]	<pre data-bbox="638 1714 1197 1854"> SET { CONSTRAINT CONSTRAINTS } { constraint [, constraint]... ALL } { IMMEDIATE DEFERRED } ;</pre>

SQL Statement	Syntax
SET ROLE	<pre>SET ROLE { role [IDENTIFIED BY password] [, role [IDENTIFIED BY password]]... ALL [EXCEPT role [, role]...] NONE } ;</pre>
SET TRANSACTION	<pre>SET TRANSACTION { { READ { ONLY WRITE } ISOLATION LEVEL { SERIALIZABLE READ COMMITTED } USE ROLLBACK SEGMENT rollback_segment } [NAME 'text'] NAME 'text' } ;</pre>
TRUNCATE	<pre>TRUNCATE { TABLE [schema.]table [{ PRESERVE PURGE } MATERIALIZED VIEW LOG] CLUSTER [schema.]cluster } [{ DROP REUSE } STORAGE] ;</pre>
UPDATE	<pre>UPDATE [hint] { dml_table_expression_clause ONLY (dml_table_expression_clause) } [t_alias] update_set_clause [where_clause] [returning_clause] ;</pre>

Table 2: Syntax for Subclauses

Subclause	Syntax
activate_standby_db_clause	<pre>ACTIVATE [PHYSICAL LOGICAL] STANDBY DATABASE [SKIP [STANDBY LOGFILE]]</pre>
add_binding_clause	<pre>ADD BINDING (parameter_type [, parameter_type]...) RETURN (return_type) [implementation_clause] using_function_clause</pre>

Subclause	Syntax
add_column_clause	<pre> ADD (column_datatype [DEFAULT expr] [{ inline_constraint [inline_constraint]... inline_ref_constraint }] [, column_datatype [DEFAULT expr] [{ inline_constraint [inline_constraint]... inline_ref_constraint }]]...) [column_properties] </pre>
add_disk_clause	<pre> ADD [FAILGROUP failgroup_name] DISK qualified_disk_clause [, qualified_disk_clause]... [[FAILGROUP failgroup_name] DISK qualified_disk_clause [, qualified_disk_clause]...]... </pre>
add_hash_index_partition	<pre> ADD PARTITION [partition_name] [TABLESPACE tablespace_name] [parallel_clause] </pre>
add_hash_partition_clause	<pre> ADD PARTITION [partition] partitioning_storage_clause [update_index_clauses] [parallel_clause] </pre>
add_hash_subpartition	<pre> ADD subpartition_spec [update_index_clauses] [parallel_clause] </pre>
add_list_partition_clause	<pre> ADD PARTITION [partition] list_values_clause [table_partition_description] [update_index_clauses] </pre>
add_list_subpartition	<pre> ADD subpartition_spec [update_index_clauses] </pre>

Subclause	Syntax
add_logfile_clauses	<pre> ADD [STANDBY] LOGFILE { [INSTANCE 'instance_name' THREAD integer] [GROUP integer] redo_log_file_spec [, [GROUP integer] redo_log_file_spec]... MEMBER 'filename' [REUSE] [, 'filename' [REUSE]]... TO logfile_descriptor [, logfile_descriptor]... } </pre>
add_overflow_clause	<pre> ADD OVERFLOW [segment_attributes_clause] [(PARTITION [segment_attributes_clause] [, PARTITION [segment_attributes_clause]...])] </pre>
add_range_partition_clause	<pre> ADD PARTITION [partition] range_values_clause [table_partition_description] [update_index_clauses] </pre>
add_table_partition	<pre> { add_range_partition_clause add_hash_partition_clause add_list_partition_clause } </pre>
alias_file_name	<pre> +diskgroup_name [(template_name)] /alias_name </pre>
allocate_extent_clause	<pre> ALLOCATE EXTENT [({ SIZE size_clause DATAFILE 'filename' INSTANCE integer } [SIZE size_clause DATAFILE 'filename' INSTANCE integer]...)] </pre>
alter_attribute_definition	<pre> { { ADD MODIFY } ATTRIBUTE { attribute [datatype] (attribute datatype [, attribute datatype]...) } DROP ATTRIBUTE </pre>

Subclause	Syntax
	<pre> { attribute (attribute [, attribute]...) } } </pre>
alter_collection_clauses	<pre> MODIFY { LIMIT integer ELEMENT TYPE datatype } </pre>
alter_datafile_clause	<pre> DATAFILE { 'filename' filenumber } [, 'filename' filenumber]... } { ONLINE OFFLINE [FOR DROP] RESIZE size_clause autoextend_clause END BACKUP } </pre>
alter_external_table_clauses	<pre> { add_column_clause modify_column_clauses drop_column_clause parallel_clause external_data_properties REJECT LIMIT { integer UNLIMITED } PROJECT COLUMN { ALL REFERENCED } } [add_column_clause modify_column_clauses drop_column_clause parallel_clause external_data_properties REJECT LIMIT { integer UNLIMITED } PROJECT COLUMN { ALL REFERENCED }]... </pre>
alter_index_partitioning	<pre> { modify_index_defaultAttrs add_hash_index_partition modify_index_partition rename_index_partition drop_index_partition split_index_partition coalesce_index_partition modify_index_subpartition } </pre>
alter_iot_clauses	<pre> { index_org_table_clause alter_overflow_clause alter_mapping_table_clauses COALESCE } </pre>

Subclause	Syntax
alter_mapping_table_clauses	<pre>MAPPING TABLE { UPDATE BLOCK REFERENCES allocate_extent_clause deallocate_unused_clause }</pre>
alter_method_spec	<pre>{ ADD DROP } { map_order_function_spec subprogram_spec } [{ ADD DROP } { map_order_function_spec subprogram_spec }]]...</pre>
alter_mv_refresh	<pre>REFRESH { { FAST COMPLETE FORCE } ON { DEMAND COMMIT } { START WITH NEXT } date WITH PRIMARY KEY USING { DEFAULT MASTER ROLLBACK SEGMENT MASTER ROLLBACK SEGMENT rollback_segment } USING { ENFORCED TRUSTED } CONSTRAINTS }</pre>
alter_overflow_clause	<pre>{ OVERFLOW { allocate_extent_clause deallocate_unused_clause } [allocate_extent_clause deallocate_unused_clause] } add_overflow_clause }</pre>
alter_session_set_clause	<pre>SET parameter_name = parameter_value [parameter_name = parameter_value]...</pre>
alter_system_reset_clause	<pre>parameter_name [SCOPE = { MEMORY SPFILE BOTH }] SID = 'sid'</pre>
alter_system_set_clause	<pre>parameter_name = parameter_value [, parameter_value]... [COMMENT 'text'] [DEFERRED]</pre>

Subclause	Syntax
	<pre>[SCOPE = { MEMORY SPFILE BOTH }] [SID = { 'sid' * }]</pre>
alter_table_partitioning	<pre>{ modify_table_defaultAttrs set_subpartition_template modify_table_partition modify_table_subpartition move_table_partition move_table_subpartition add_table_partition coalesce_table_partition drop_table_partition drop_table_subpartition rename_partition_subpart truncate_partition_subpart split_table_partition split_table_subpartition merge_table_partitions merge_table_subpartitions exchange_partition_subpart }</pre>
alter_table_properties	<pre>{ { physical_attributes_clause logging_clause table_compression supplemental_table_logging allocate_extent_clause deallocate_unused_clause shrink_clause { CACHE NOCACHE } upgrade_table_clause records_per_block_clause parallel_clause row_movement_clause } [physical_attributes_clause logging_clause table_compression supplemental_table_logging allocate_extent_clause deallocate_unused_clause shrink_clause { CACHE NOCACHE } upgrade_table_clause records_per_block_clause parallel_clause row_movement_clause] ... RENAME TO new_table_name } [alter_iot_clauses]</pre>

Subclause	Syntax
alter_tempfile_clause	<pre> TEMPFILE { 'filename' [, 'filename']... filenumber [, filenumber]... } { RESIZE size_clause autoextend_clause DROP [INCLUDING DATAFILES] ONLINE OFFLINE } </pre>
alter_varray_col_properties	<pre> MODIFY VARRAY varray_item (modify_LOB_parameters) </pre>
analytic_clause	<pre> [query_partition_clause] [order_by_clause [windowing_clause]] </pre>
archive_log_clause	<pre> ARCHIVE LOG [INSTANCE 'instance_name' THREAD integer] { { SEQUENCE integer CHANGE integer CURRENT [NOSWITCH] GROUP integer LOGFILE 'filename' [USING BACKUP CONTROLFILE] NEXT ALL START } [TO 'location'] STOP } </pre>
array_DML_clause	<pre> [WITH WITHOUT] ARRAY DML [([schema.]type [, [schema.]varray_type]) [, ([schema.]type [, [schema.]varray_type])...] </pre>
attribute_clause	<pre> ATTRIBUTE level DETERMINES { dependent_column (dependent_column [, dependent_column]...) } </pre>
auditing_by_clause	<pre> BY { proxy [, proxy]... user [, user]... } </pre>

Subclause	Syntax
auditing_on_clause	ON { [schema.]object DIRECTORY directory_name DEFAULT }
autoextend_clause	AUTOEXTEND { OFF ON [NEXT size_clause] [maxsize_clause] }
binding_clause	BINDING (parameter_type [, parameter_type]...) RETURN return_type [implementation_clause] using_function_clause , (parameter_type [, parameter_type]...) RETURN return_type [implementation_clause] using_function_clause]...
bitmap_join_index_clause	[schema.]table ([[schema.]table. t_alias.]column [ASC DESC] [, [[schema.]table. t_alias.]column [ASC DESC]]...) FROM [schema.]table [t_alias] [, [schema.]table [t_alias]]... WHERE condition [local_partitioned_index] index_attributes
build_clause	BUILD { IMMEDIATE DEFERRED }
C_declaration	C [NAME name] LIBRARY lib_name [AGENT IN (argument[, argument]...)] [WITH CONTEXT] [PARAMETERS (parameter[, parameter]...)]
call_spec	LANGUAGE { Java_declaration C_declaration }
cancel_clause	CANCEL [IMMEDIATE] [WAIT NOWAIT]

Subclause	Syntax
cell_assignment	<pre> measure_column [{ condition expr single_column_for_loop } [, { condition expr single_column_for_loop }]]... multi_column_for_loop }] </pre> <p>Note: The outer square brackets are part of the syntax. In this case, they do not indicate optionality.</p>
cell_reference_options	<pre> [{ IGNORE KEEP } NAV] [UNIQUE { DIMENSION SINGLE REFERENCE }] </pre>
character_set_clause	CHARACTER SET character_set
check_datafiles_clause	CHECK DATAFILES [GLOBAL LOCAL]
check_diskgroup_clauses	<pre> CHECK { ALL DISK disk_name [, disk_name]... DISKS IN FAILGROUP failgroup_name [, failgroup_name]... FILE filename [, filename]... } [CHECK { ALL DISK disk_name [, disk_name]... DISKS IN FAILGROUP failgroup_name [, failgroup_name]... FILE filename [, filename]... }]... [REPAIR NOREPAIR] </pre>

Subclause	Syntax
checkpoint_clause	CHECKPOINT [GLOBAL LOCAL]
cluster_index_clause	CLUSTER [schema.] cluster_index_attributes
coalesce_index_partition	COALESCE PARTITION [parallel_clause]
coalesce_table_partition	COALESCE PARTITION [update_index_clauses] [parallel_clause]
column_association	COLUMNS [schema.]table.column [, [schema.]table.column]... using_statistics_type
column_clauses	{ add_column_clause modify_column_clause drop_column_clause } [add_column_clause modify_column_clause drop_column_clause]... rename_column_clause modify_collection_retrieval [modify_collection_retrieval]... modify_LOB_storage_clause alter_varray_col_properties }
column_properties	{ object_type_col_properties nested_table_col_properties { varray_col_properties LOB_storage_clause } (LOB_partition_storage [, LOB_partition_storage]...) } XMLType_column_properties } [{ object_type_col_properties nested_table_col_properties { varray_col_properties LOB_storage_clause } (LOB_partition_storage [, LOB_partition_storage]...) } XMLType_column_properties }]...

Subclause	Syntax
commit_switchover_clause	<pre> { PREPARE COMMIT } TO SWITCHOVER [TO { { PHYSICAL LOGICAL } PRIMARY PHYSICAL STANDBY [{ WITH WITHOUT } SESSION SHUTDOWN { WAIT NOWAIT }] LOGICAL STANDBY } CANCEL] </pre>
compile_type_clause	<pre> COMPILE [DEBUG] [SPECIFICATION BODY] [compiler_parameters_clause [compiler_parameters_clause] ...] [REUSE SETTINGS] </pre>
compiler_parameters_clause	parameter_name = parameter_value
composite_partitioning	<pre> PARTITION BY RANGE (column_list) [subpartition_by_list subpartition_by_hash] (PARTITION [partition] range_values_clause table_partition_description [, PARTITION [partition] range_values_clause table_partition_description] ...) </pre>
compute_statistics_clause	COMPUTE [SYSTEM] STATISTICS [for_clause]
conditional_insert_clause	<pre> [ALL FIRST] WHEN condition THEN insert_into_clause [values_clause] [error_logging_clause] [insert_into_clause [values_clause] [error_logging_clause]]... [WHEN condition THEN insert_into_clause [values_clause] [error_logging_clause] [insert_into_clause [values_clause] [error_logging_clause]]...]... [ELSE insert_into_clause </pre>

Subclause	Syntax
	<pre> [values_clause] [error_logging_clause] [insert_into_clause [values_clause] [error_logging_clause]] ...]</pre>
constraint	<pre> { inline_constraint out_of_line_constraint inline_ref_constraint out_of_line_ref_constraint }</pre>
constraint_clauses	<pre> { ADD { out_of_line_constraint [out_of_line_constraint]... out_of_line_REF_constraint } MODIFY { CONSTRAINT constraint PRIMARY KEY UNIQUE (column [, column]...) } constraint_state RENAME CONSTRAINT old_name TO new_name drop_constraint_clause }</pre>
constraint_state	<pre> [[[NOT] DEFERRABLE] INITIALLY { IMMEDIATE DEFERRED }] [INITIALLY { IMMEDIATE DEFERRED }] [[NOT] DEFERRABLE]] [RELY NORELY] [using_index_clause] [ENABLE DISABLE] [VALIDATE NOVALIDATE] [exceptions_clause]</pre>
constructor_declaration	<pre> [FINAL] [INSTANTIABLE] CONSTRUCTOR FUNCTION datatype [[SELF IN OUT datatype, parameter datatype [, parameter datatype]...] RETURN SELF AS RESULT { IS AS } { pl/sql_block call_spec }</pre>
constructor_spec	<pre> [FINAL] [INSTANTIABLE] CONSTRUCTOR FUNCTION datatype [([SELF IN OUT datatype,]</pre>

Subclause	Syntax
	<pre> parameter datatype [, parameter datatype]...)] RETURN SELF AS RESULT [{ IS AS } call_spec] </pre>
context_clause	<pre> [WITH INDEX CONTEXT, SCAN CONTEXT implementation_type [COMPUTE ANCILLARY DATA]] [WITH COLUMN CONTEXT] </pre>
controlfile_clauses	<pre> { CREATE [LOGICAL PHYSICAL] STANDBY CONTROLFILE AS 'filename' [REUSE] BACKUP CONTROLFILE TO { 'filename' [REUSE] trace_file_clause } } </pre>
create_datafile_clause	<pre> CREATE DATAFILE { 'filename' filenumber } [, 'filename' filenumber]... } [AS { file_specification [, file_specification]... NEW }] </pre>
create_incomplete_type	<pre> CREATE [OR REPLACE] TYPE [schema.]type_name ; </pre>
create_mv_refresh	<pre> { REFRESH { { FAST COMPLETE FORCE } ON { DEMAND COMMIT } { START WITH NEXT } date WITH { PRIMARY KEY ROWID } USING { DEFAULT [MASTER LOCAL] ROLLBACK SEGMENT [MASTER LOCAL] ROLLBACK SEGMENT rollback_segment } [DEFAULT [MASTER LOCAL] ROLLBACK SEGMENT [MASTER LOCAL] ROLLBACK SEGMENT rollback_segment]... </pre>

Subclause	Syntax
	<pre> USING { ENFORCED TRUSTED } CONSTRAINTS } [{ FAST COMPLETE FORCE } ON { DEMAND COMMIT } { START WITH NEXT } date WITH { PRIMARY KEY ROWID } USING { DEFAULT [MASTER LOCAL] ROLLBACK SEGMENT [MASTER LOCAL] ROLLBACK SEGMENT rollback_segment } [DEFAULT [MASTER LOCAL] ROLLBACK SEGMENT [MASTER LOCAL] ROLLBACK SEGMENT rollback_segment] ... USING { ENFORCED TRUSTED } CONSTRAINTS] ... NEVER REFRESH }</pre>
create_nested_table_type	<pre>CREATE [OR REPLACE] TYPE [schema.]type_name [OID 'object_identifier'] { IS AS } TABLE OF datatype ;</pre>
create_object_type	<pre>CREATE [OR REPLACE] TYPE [schema.]type_name [OID 'object_identifier'] [invoker_rights_clause] { { IS AS } OBJECT UNDER [schema.]supertype } [sqlj_object_type] [(attribute datatype [sqlj_object_type_attr] [, attribute datatype [sqlj_object_type_attr]... [, element_spec [, element_spec]...])] [[NOT] FINAL] [[NOT] INSTANTIABLE] ;</pre>

Subclause	Syntax
create_varray_type	<pre>CREATE [OR REPLACE] TYPE [schema.]type_name [OID 'object_identifier'] { IS AS } { VARRAY VARYING ARRAY } (limit) OF datatype ;</pre>
database_file_clauses	<pre>{ RENAME FILE 'filename' [, 'filename']... TO 'filename' create_datafile_clause alter_datafile_clause alter_tempfile_clause }</pre>
database_logging_clauses	<pre>{ LOGFILE [GROUP integer] file_specification [, [GROUP integer] file_specification]... MAXLOGFILES integer MAXLOGMEMBERS integer MAXLOGHISTORY integer { ARCHIVELOG NOARCHIVELOG } FORCE LOGGING }</pre>
datafile_tempfile_clauses	<pre>{ ADD { DATAFILE TEMPFILE } [file_specification [, file_specification]...] RENAME DATAFILE 'filename' [, 'filename']... TO 'filename' [, 'filename']... { DATAFILE TEMPFILE } { ONLINE OFFLINE } }</pre>
datafile_tempfile_spec	<pre>['filename'] [SIZE size_clause] [REUSE] [autoextend_clause]</pre>
dblink	<pre>database[.domain [.domain]...] [@ connect_descriptor]</pre>
dblink_authentication	<pre>AUTHENTICATED BY user IDENTIFIED BY password</pre>
deallocate_unused_clause	<pre>DEALLOCATE UNUSED [KEEP size_clause]</pre>

Subclause	Syntax
default_cost_clause	DEFAULT COST (cpu_cost, io_cost, network_cost)
default_selectivity_clause	DEFAULT SELECTIVITY default_selectivity
default_tablespace	DEFAULT TABLESPACE tablespace [DATAFILE datafile_tempfile_spec] extent_management_clause
default_settings_clauses	{ SET DEFAULT { BIGFILE SMALLFILE } TABLESPACE DEFAULT TABLESPACE tablespace DEFAULT TEMPORARY TABLESPACE { tablespace tablespace_group_name } RENAME GLOBAL_NAME TO database.domain [.domain]... { ENABLE BLOCK CHANGE TRACKING [USING FILE 'filename' [REUSE]] DISABLE BLOCK CHANGE TRACKING } flashback_mode_clause set_time_zone_clause }
default_temp_tablespace	[BIGFILE SMALLFILE] DEFAULT TEMPORARY TABLESPACE tablespace [TEMPFILE file_specification [, file_specification]...] extent_management_clause
dependent_handling_clause	{ INVALIDATE CASCADE [{ [NOT] INCLUDING TABLE DATA CONVERT TO SUBSTITUTABLE }] [[FORCE] exceptions_clause] }
dimension_join_clause	JOIN KEY { child_key_column (child_key_column [, child_key_column]...) } REFERENCES parent_level [JOIN KEY { child_key_column (child_key_column [, child_key_column]...) } REFERENCES parent_level]

Subclause	Syntax
disk_clauses	<pre> { diskgroup_name { add_disk_clause drop_disk_clauses resize_disk_clauses } { diskgroup_name ALL } undrop_disk_clause } </pre>
diskgroup_alias_clauses	<pre> { ADD ALIAS alias_name FOR filename [, alias_name FOR filename]... DROP ALIAS alias_name [, alias_name]... RENAME ALIAS old_alias_name TO new_alias_name [, old_alias_name TO new_alias_name]... } </pre>
diskgroup_availability	<pre> { MOUNT DISMOUNT [FORCE NOFORCE] } </pre>
diskgroup_clauses	<pre> { diskgroup_name { rebalance_diskgroup_clause check_diskgroup_clauses diskgroup_template_clauses diskgroup_directory_clauses diskgroup_alias_clauses drop_diskgroup_file_clause } { diskgroup_name ALL } diskgroup_availability } </pre>
diskgroup_directory_clauses	<pre> { ADD DIRECTORY filename [, filename]... DROP DIRECTORY filename [FORCE NOFORCE] [, filename [FORCE NOFORCE]]... RENAME DIRECTORY old_dir_name TO new_dir_name [, old_dir_name TO new_dir_name]... } </pre>
diskgroup_file_spec	<pre> [' { fully_qualified_file_name numeric_file_name incorporate_file_name alias_file_name } } </pre>

Subclause	Syntax
	<pre>] [SIZE size_clause] [REUSE] [autoextend_clause] </pre>
diskgroup_template_clauses	<pre> { { ADD ALTER } TEMPLATE qualified_template_clause [, qualified_template_clause]... DROP TEMPLATE template_name [, template_name]... } </pre>
distributed_recov_clauses	<pre>{ ENABLE DISABLE } DISTRIBUTED RECOVERY</pre>
dml_event_clause	<pre> { DELETE INSERT UPDATE [OF column [, column]...] } [OR { DELETE INSERT UPDATE [OF column [, column]...] }] [referencing_clause] [FOR EACH ROW] </pre>
dml_table_expression_clause	<pre> { [schema.] table [{ PARTITION (partition) SUBPARTITION (subpartition) } @ dblink] { view materialized view } [@ dblink] } (subquery [subquery_restriction_clause]) table_collection_expression } </pre>
domain_index_clause	<pre>INDEXTYPE IS indextype [parallel_clause] [PARAMETERS ('ODCI_parameters')]</pre>
drop_binding_clause	<pre>DROP BINDING (parameter_type [, parameter_type]...) [FORCE]</pre>

Subclause	Syntax
drop_column_clause	<pre> { SET UNUSED { COLUMN column (column [, column]...) } [{ CASCADE CONSTRAINTS INVALIDATE } [CASCADE CONSTRAINTS INVALIDATE]...] DROP { COLUMN column (column [, column]...) } [{ CASCADE CONSTRAINTS INVALIDATE } [CASCADE CONSTRAINTS INVALIDATE]...] [CHECKPOINT integer] DROP { UNUSED COLUMNS COLUMNS CONTINUE } [CHECKPOINT integer] } </pre>
drop_constraint_clause	<pre> DROP { { PRIMARY KEY UNIQUE (column [, column]...) } [CASCADE] [{ KEEP DROP } INDEX] CONSTRAINT constraint [CASCADE] } </pre>
drop_disk_clauses	<pre> DROP { DISK disk_name [FORCE NOFORCE] [, disk_name [FORCE NOFORCE]]... DISKS IN FAILGROUP failgroup_name [FORCE NOFORCE] [, failgroup_name [FORCE NOFORCE]]... } </pre>
drop_diskgroup_file_clause	<pre> DROP FILE filename [, filename]... </pre>
drop_index_partition	<pre> DROP PARTITION partition_name </pre>
drop_logfile_clauses	<pre> DROP [STANDBY] LOGFILE { logfile_descriptor [, logfile_descriptor]... MEMBER 'filename' [, 'filename']... } </pre>

Subclause	Syntax
drop_table_partition	DROP PARTITION partition [update_index_clauses [parallel_clause]]
drop_table_subpartition	DROP SUBPARTITION subpartition [update_index_clauses [parallel_clause]]
element_spec	[inheritance_clauses] { subprogram_spec constructor_spec map_order_function_spec } [subprogram_clause constructor_spec map_order_function_spec]... [, pragma_clause]
else_clause	ELSE else_expr
enable_disable_clause	{ ENABLE DISABLE } [VALIDATE NOVALIDATE] { UNIQUE (column [, column]...) PRIMARY KEY CONSTRAINT constraint } [using_index_clause] [exceptions_clause] [CASCADE] [{ KEEP DROP } INDEX]
end_session_clauses	{ DISCONNECT SESSION 'integer1, integer2' [POST_TRANSACTION] KILL SESSION 'integer1, integer2' } [IMMEDIATE]
estimate_statistics_clause	ESTIMATE [SYSTEM] STATISTICS [for_clause] [SAMPLE integer { ROWS PERCENT }]
exceptions_clause	EXCEPTIONS INTO [schema.]table
exchange_partition_subpart	EXCHANGE { PARTITION partition SUBPARTITION subpartition } WITH TABLE table [{ INCLUDING EXCLUDING } INDEXES] [{ WITH WITHOUT } VALIDATION] [exceptions_clause] [update_index_clauses [parallel_clause]]

Subclause	Syntax
expr	{ simple_expression compound_expression case_expression cursor_expression datetime_expression function_expression interval_expression object_access_expression scalar_subquery_expression model_expression type_constructor_expression variable_expression }
expression_list	{ expr [, expr]... (expr [, expr]...) }
extended_attribute_clause	ATTRIBUTE attribute LEVEL level DETERMINES { dependent_column (dependent_column [, dependent_column]...) [LEVEL level DETERMINES { dependent_column (dependent_column [, dependent_column]...)]...]
extent_management_clause	EXTENT MANAGEMENT { DICTIONARY LOCAL AUTOALLOCATE UNIFORM SIZE size_clause } }
external_data_properties	DEFAULT DIRECTORY directory [ACCESS PARAMETERS { (opaque_format_spec) USING CLOB subquery }] LOCATION ([directory:] 'location_specifier' [, [directory:] 'location_specifier']...)

Subclause	Syntax
external_table_clause	([TYPE access_driver_type] external_data_properties) [REJECT LIMIT { integer UNLIMITED }]
file_specification	{ datafile_tempfile_spec diskgroup_file_spec redo_log_file_spec }
finish_clause	[DISCONNECT [FROM SESSION]] [parallel_clause] FINISH [SKIP [STANDBY LOGFILE]] [WAIT NOWAIT]
flashback_mode_clause	FLASHBACK { ON OFF }
flashback_query_clause	[VERSIONS BETWEEN { SCN TIMESTAMP } { expr MINVALUE } AND { expr MAXVALUE }] AS OF { SCN TIMESTAMP } expr
for_clause	FOR { TABLE ALL [INDEXED] COLUMNS [SIZE integer] COLUMNS [SIZE integer] { column attribute } [SIZE integer] [{ column attribute } [SIZE integer]]]... ALL [LOCAL] INDEXES } [FOR { TABLE ALL [INDEXED] COLUMNS [SIZE integer] COLUMNS [SIZE integer] { column attribute } [SIZE integer] [{ column attribute } [SIZE integer]]]... ALL [LOCAL] INDEXES }]...
for_update_clause	FOR UPDATE [OF [[schema.] { table view } .] column [, [[schema.]

Subclause	Syntax
	<pre> { table view } .]column]...] [NOWAIT WAIT integer] </pre>
full_database_recovery	<pre> [STANDBY] DATABASE [{ UNTIL { CANCEL TIME date CHANGE integer } USING BACKUP CONTROLFILE } [UNTIL { CANCEL TIME date CHANGE integer } USING BACKUP CONTROLFILE]]...] </pre>
fully_qualified_file_name	+diskgroup_name/db_name/file_type/ file_type_tag.filenumbers.incarnation_number
function_association	<pre> { FUNCTIONS [schema.]function [, [schema.]function]... PACKAGES [schema.]package [, [schema.]package]... TYPES [schema.]type [, [schema.]type]... INDEXES [schema.]index [, [schema.]index]... INDEXTYPES [schema.]indextype [, [schema.]indextype]... } { using_statistics_type { default_cost_clause [, default_selectivity_clause] default_selectivity_clause [, default_cost_clause] } } </pre>
function_declaration	<pre> FUNCTION name (parameter datatype[, parameter datatype]...) RETURN datatype { IS AS } { pl/sql_block call_spec } </pre>

Subclause	Syntax
function_spec	<pre>FUNCTION name (parameter datatype [, parameter datatype]]...) return_clause</pre>
general_recovery	<pre>RECOVER [AUTOMATIC] [FROM 'location'] { full_database_recovery partial_database_recovery LOGFILE 'filename' } [{ TEST ALLOW integer CORRUPTION parallel_clause } [TEST ALLOW integer CORRUPTION parallel_clause]...] CONTINUE [DEFAULT] CANCEL }</pre>
global_partitioned_index	<pre>GLOBAL PARTITION BY { RANGE (column_list) (index_partitioning_clause) HASH (column_list) { individual_hash_partitions hash_partitions_by_quantity } }</pre>
grant_object_privileges	<pre>{ object_privilege ALL [PRIVILEGES] } [(column [, column]...)] [{ object_privilege ALL [PRIVILEGES] } [(column [, column]...)]]... on_object_clause TO grantee_clause [WITH HIERARCHY OPTION] [WITH GRANT OPTION]</pre>
grant_system_privileges	<pre>{ system_privilege role ALL PRIVILEGES } [, { system_privilege</pre>

Subclause	Syntax
	<pre> role ALL PRIVILEGES }]... TO grantee_clause [IDENTIFIED BY password] [WITH ADMIN OPTION] </pre>
grantee_clause	<pre> { user role PUBLIC } [, { user role PUBLIC }]... </pre>
group_by_clause	<pre> GROUP BY { expr rollup_cube_clause grouping_sets_clause } [, { expr rollup_cube_clause grouping_sets_clause }]]... [HAVING condition] </pre>
grouping_expression_list	<pre> expression_list [, expression_list]... </pre>
grouping_sets_clause	<pre> GROUPING SETS ({ rollup_cube_clause grouping_expression_list }) </pre>
hash_partitioning	<pre> PARTITION BY HASH (column [, column] ...) { individual_hash_partitions hash_partitions_by_quantity } </pre>
hash_partitions_by_quantity	<pre> PARTITIONS hash_partition_quantity [STORE IN (tablespace [, tablespace]...)] [OVERFLOW STORE IN (tablespace [, tablespace]...)] </pre>
hierarchical_query_clause	<pre> [START WITH condition] CONNECT BY [NOCYCLE] condition </pre>
hierarchy_clause	<pre> HIERARCHY hierarchy (child_level CHILD OF parent_level [CHILD OF parent_level]... [dimension_join_clause]) </pre>

Subclause	Syntax
implementation_clause	{ ANCILLARY TO primary_operator (parameter_type [, parameter_type]...) [, primary_operator (parameter_type [, parameter_type]]...)]... context_clause }
incomplete_file_name	+diskgroup_name [(template_name)]
index_attributes	[{ physical_attributes_clause logging_clause ONLINE COMPUTE STATISTICS TABLESPACE { tablespace DEFAULT } key_compression { SORT NOSORT } REVERSE parallel_clause } [physical_attributes_clause logging_clause ONLINE COMPUTE STATISTICS TABLESPACE { tablespace DEFAULT } key_compression { SORT NOSORT } REVERSE parallel_clause]...]
index_expr	{ column column_expression }
index_org_overflow_clause	[INCLUDING column_name] OVERFLOW [segment_attributes_clause]
index_org_table_clause	[{ mapping_table_clause PCTTHRESHOLD integer key_compression } [mapping_table_clause PCTTHRESHOLD integer key_compression]...] [index_org_overflow_clause]

Subclause	Syntax
index_partition_description	<pre> PARTITION [partition [{ segment_attributes_clause key_compression } [segment_attributes_clause key_compression]...]] </pre>
index_partitioning_clause	<pre> PARTITION [partition] VALUES LESS THAN (value[, value...]) [segment_attributes_clause] </pre>
index_properties	<pre> [{ { global_partitioned_index local_partitioned_index } index_attributes } [{ { global_partitioned_index local_partitioned_index } index_attributes }]... domain_index_clause] </pre>
index_subpartition_clause	<pre> { STORE IN (tablespace[, tablespace]...) (SUBPARTITION [subpartition [TABLESPACE tablespace]] [, SUBPARTITION [subpartition [TABLESPACE tablespace]]]...) } </pre>
individual_hash_partitions	<pre> (PARTITION [partition partitioning_storage_clause] [, PARTITION [partition partitioning_storage_clause]]...) </pre>
inheritance_clauses	<pre> [NOT] { OVERRIDING FINAL INSTANTIABLE } [[NOT] { OVERRIDING FINAL INSTANTIABLE }]... </pre>

Subclause	Syntax
inline_constraint	[CONSTRAINT constraint_name] { [NOT] NULL UNIQUE PRIMARY KEY references_clause CHECK (condition) } [constraint_state]
inline_ref_constraint	{ SCOPE IS [schema.] scope_table WITH ROWID [CONSTRAINT constraint_name] references_clause [constraint_state] }
inner_cross_join_clause	table_reference { [INNER] JOIN table_reference { ON condition USING (column [, column]...) } { CROSS NATURAL [INNER] } JOIN table_reference }
insert_into_clause	INTO dml_table_expression_clause [t_alias] [(column [, column]...)]
integer	[+ -] digit [digit]...
interval_day_to_second	INTERVAL '{ integer integer time_expr time_expr }' { { DAY HOUR MINUTE } [(leading_precision)] SECOND [(leading_precision [, fractional_seconds_precision])] } [TO { DAY HOUR MINUTE SECOND [(fractional_seconds_precision)] }]
interval_year_to_month	INTERVAL 'integer [- integer]' { YEAR MONTH } [(precision)] [TO { YEAR MONTH }]

Subclause	Syntax
into_clause	INTO [schema.] table
invoker_rights_clause	AUTHID { CURRENT_USER DEFINER }
Java_declaration	JAVA NAME 'string'
join_clause	{ inner_cross_join_clause outer_join_clause }
key_compression	{ COMPRESS [integer] NOCOMPRESS }
level_clause	LEVEL level IS { level_table.level_column (level_table.level_column [, level_table.level_column]...) }
list_partitioning	PARTITION BY LIST (column) (PARTITION [partition] list_values_clause table_partition_description , PARTITION [partition] list_values_clause table_partition_description]...)
list_values_clause	VALUES ({ value NULL [, { value NULL }...] DEFAULT)
LOB_parameters	{ TABLESPACE tablespace { ENABLE DISABLE } STORAGE IN ROW storage_clause CHUNK integer PCTVERSION integer RETENTION FREEPOOLS integer { CACHE { NOCACHE CACHE READS } [logging_clause] } } [TABLESPACE tablespace { ENABLE DISABLE } STORAGE IN ROW storage_clause

Subclause	Syntax
	<pre> CHUNK integer PCTVERSION integer RETENTION FREEPOOLS integer { CACHE { NOCACHE CACHE READS } [logging_clause] }] ... </pre>
LOB_partition_storage	<pre> PARTITION partition { LOB_storage_clause varray_col_properties } [LOB_storage_clause varray_col_properties] ... [(SUBPARTITION subpartition { LOB_storage_clause varray_col_properties } [LOB_storage_clause varray_col_properties] ...)] </pre>
LOB_storage_clause	<pre> LOB { (LOB_item [, LOB_item]...) STORE AS (LOB_parameters) (LOB_item) STORE AS { LOB_segnname (LOB_parameters) LOB_segnname (LOB_parameters) } } </pre>
local_partitioned_index	<pre> LOCAL [on_range_partitioned_table on_list_partitioned_table on_hash_partitioned_table on_comp_partitioned_table] </pre>
logfile_clause	<pre> LOGFILE [GROUP integer] file_specification [, [GROUP integer] file_specification]... </pre>
logfile_clauses	<pre> { { ARCHIVELOG [MANUAL] NOARCHIVELOG } [NO] FORCE LOGGING RENAME FILE 'filename' [, 'filename']... TO 'filename' </pre>

Subclause	Syntax
	<pre> CLEAR [UNARCHIVED] LOGFILE logfile_descriptor [, logfile_descriptor]... [UNRECOVERABLE DATAFILE] add_logfile_clauses drop_logfile_clauses supplemental_db_logging } </pre>
logfile_descriptor	<pre> { GROUP integer ('filename' [, 'filename']...) 'filename' } </pre>
logging_clause	{ LOGGING NOLOGGING }
main_model	<pre> [MAIN main_model_name] model_column_clauses [cell_reference_options] model_rules_clause </pre>
managed_standby_recovery	<pre> RECOVER MANAGED STANDBY DATABASE [recover_clause cancel_clause finish_clause] </pre>
map_order_func_declaration	{ MAP ORDER } MEMBER function_declaration
map_order_function_spec	{ MAP ORDER } MEMBER function_spec
mapping_table_clauses	{ MAPPING TABLE NOMAPPING }
materialized_view_props	<pre> [column_properties] [table_partitioning_clauses] [CACHE NOCACHE] [parallel_clause] [build_clause] </pre>
maximize_standby_db_clause	<pre> SET STANDBY DATABASE TO MAXIMIZE { PROTECTION AVAILABILITY PERFORMANCE } </pre>
maxsize_clause	MAXSIZE { UNLIMITED size_clause }
merge_insert_clause	<pre> WHEN NOT MATCHED THEN INSERT [(column [, column]...)] VALUES ({ expr [, expr]... DEFAULT }) [where_clause] </pre>

Subclause	Syntax
merge_table_partitions	MERGE PARTITIONS partition_1, partition_2 [INTO partition_spec] [update_index_clauses] [parallel_clause]
merge_table_subpartitions	MERGE SUBPARTITIONS subpart_1, subpart_2 [INTO subpartition_spec] [update_index_clauses] [parallel_clause]
merge_update_clause	WHEN MATCHED THEN UPDATE SET column = { expr DEFAULT } [, column = { expr DEFAULT }]... [where_clause] [DELETE where_clause]
model_clause	MODEL [cell_reference_options] [return_rows_clause] [reference_model] [reference_model]... main_model
model_column	expr [[AS] c_alias]
model_column_clauses	[query_partition_clause [c_alias]] DIMENSION BY (model_column [, model_column]...) MEASURES (model_column [, model_column]...)
model_rules_clause	RULES [UPSERT UPDATE] [{ AUTOMATIC SEQUENTIAL } ORDER] [ITERATE (number) [UNTIL (condition)]] [[UPDATE UPSERT] cell_assignment [order_by_clause] = expr [[UPDATE UPSERT] cell_assignment [order_by_clause] = expr]...]
modify_col_properties	(column [datatype] [DEFAULT expr] [inline_constraint [inline_constraint]...] [LOB_storage_clause] [, column [datatype] [DEFAULT expr] [inline_constraint

Subclause	Syntax
	<pre> [inline_constraint]...] [LOB_storage_clause]]) </pre>
modify_col_substitutable	<pre> COLUMN column [NOT] SUBSTITUTABLE AT ALL LEVELS [FORCE] </pre>
modify_collection_retrieval	<pre> MODIFY NESTED TABLE collection_item RETURN AS { LOCATOR VALUE } </pre>
modify_column_clauses	<pre> MODIFY { modify_col_properties modify_col_substitutable } </pre>
modify_hash_partition	<pre> MODIFY PARTITION partition { partition_attributes alter_mapping_table_clause [REBUILD] UNUSABLE LOCAL INDEXES } </pre>
modify_hash_subpartition	<pre> { { allocate_extent_clause deallocate_unused_clause shrink_clause { LOB LOB_item VARRAY varray } } modify_LOB_parameters [{ LOB LOB_item VARRAY varray } modify_LOB_parameters]... } [REBUILD] UNUSABLE LOCAL INDEXES } </pre>
modify_index_default_attrs	<pre> MODIFY DEFAULT ATTRIBUTES [FOR PARTITION partition] { physical_attributes_clause TABLESPACE { tablespace DEFAULT } logging_clause } [physical_attributes_clause TABLESPACE { tablespace DEFAULT } logging_clause]... </pre>

Subclause	Syntax
modify_index_partition	<pre> MODIFY PARTITION partition { { deallocate_unused_clause allocate_extent_clause physical_attributes_clause logging_clause key_compression } [deallocate_unused_clause allocate_extent_clause physical_attributes_clause logging_clause key_compression]... PARAMETERS ('ODCI_parameters') COALESCE UPDATE BLOCK REFERENCES UNUSABLE } </pre>
modify_index_subpartition	<pre> MODIFY SUBPARTITION subpartition { UNUSABLE allocate_extent_clause deallocate_unused_clause } </pre>
modify_list_partition	<pre> MODIFY PARTITION partition { partition_attributes {ADD DROP} VALUES (partition_value[, partition_value]...) [REBUILD] UNUSABLE LOCAL INDEXES } </pre>
modify_list_subpartition	<pre> { allocate_extent_clause deallocate_unused_clause shrink_clause { LOB LOB_item VARRAY varray } modify_LOB_parameters [{ LOB LOB_item VARRAY varray } modify_LOB_parameters] ... [REBUILD] UNUSABLE LOCAL INDEXES { ADD DROP } VALUES (value[, value]...) } </pre>
modify_LOB_parameters	<pre> { storage_clause PCTVERSION integer RETENTION FREEPOOLS integer REBUILD FREEPOOLS { CACHE { NOCACHE CACHE READS } [logging_clause] } </pre>

Subclause	Syntax
	<pre> } allocate_extent_clause deallocate_unused_clause shrink_clause } [storage_clause PCTVERSION integer RETENTION FREEPOOLS integer REBUILD FREEPOOLS { CACHE { NOCACHE CACHE READS } [logging_clause] } allocate_extent_clause deallocate_unused_clause shrink_clause] ... </pre>
modify_LOB_storage_clause	<pre>MODIFY LOB (LOB_item) (modify_LOB_parameters)</pre>
modify_range_partition	<pre>MODIFY PARTITION partition { partition_attributes { add_hash_subpartition add_list_subpartition } COALESCE SUBPARTITION [update_index_clauses] [parallel_clause] alter_mapping_table_clause [REBUILD] UNUSABLE LOCAL INDEXES }</pre>
modify_table_default_attrs	<pre>MODIFY DEFAULT ATTRIBUTES [FOR PARTITION partition] [segment_attributes_clause] [table_compression] [PCTTHRESHOLD integer] [key_compression] [alter_overflow_clause] [{ LOB (LOB_item) VARRAY varray } (LOB_parameters) [{ LOB (LOB_item) VARRAY varray } (LOB_parameters)] ...]</pre>

Subclause	Syntax
modify_table_partition	{ modify_range_partition modify_hash_partition modify_list_partition }
modify_table_subpartition	MODIFY SUBPARTITION subpartition { modify_hash_subpartition modify_list_subpartition }
move_table_clause	MOVE [ONLINE] [segment_attributes_clause] [table_compression] [index_org_table_clause] [{ LOB_storage_clause varray_col_properties } [{ LOB_storage_clause varray_col_properties }] ...] [parallel_clause]
move_table_partition	MOVE PARTITION partition [MAPPING TABLE] [table_partition_description] [update_index_clauses] [parallel_clause]
move_table_subpartition	MOVE SUBPARTITION subpartition_spec [update_index_clauses] [parallel_clause]
multi_column_for_loop	FOR (dimension_column [, dimension_column]...) IN ({ (literal [, literal]...) [(literal [, literal]...)...] subquery })
multi_table_insert	{ ALL insert_into_clause [values_clause] [insert_into_clause [values_clause]]... conditional_insert_clause } subquery

Subclause	Syntax
multiset_except	nested_table1 MULTISET EXCEPT [ALL DISTINCT] nested_table2
multiset_intersect	nested_table1 MULTISET INTERSECT [ALL DISTINCT] nested_table2
multiset_union	nested_table1 MULTISET UNION [ALL DISTINCT] nested_table2
nested_table_col_properties	NESTED TABLE { nested_item COLUMN_VALUE } [substitutable_column_clause] STORE AS storage_table [({ object_properties) physical_properties] column_properties] } [(object_properties) physical_properties] column_properties]]...)] [RETURN AS { LOCATOR VALUE }]
new_values_clause	{ INCLUDING EXCLUDING } NEW VALUES
number	[+ -] { digit [digit]... [.] [digit [digit]...] . digit [digit]... } [e [+ -] digit [digit]...] [f d]
numeric_file_name	+diskgroup_name.filenumbers.incarnation_number
object_properties	{ { column attribute } [DEFAULT expr] [inline_constraint [inline_constraint]... inline_ref_constraint] { out_of_line_constraint out_of_line_ref_constraint supplemental_logging_props } }

Subclause	Syntax
object_table	<pre>CREATE [GLOBAL TEMPORARY] TABLE [schema.]table OF [schema.]object_type [object_table_substitution] [(object_properties)] [ON COMMIT { DELETE PRESERVE } ROWS] [OID_clause] [OID_index_clause] [physical_properties] [table_properties] ;</pre>
object_table_substitution	[NOT] SUBSTITUTABLE AT ALL LEVELS
object_type_col_properties	COLUMN column substitutable_column_clause
object_view_clause	<pre>OF [schema.]type_name { WITH OBJECT IDENTIFIER { DEFAULT (attribute [, attribute]...) } UNDER [schema.]superview } ({ out_of_line_constraint attribute inline_constraint [inline_constraint]... } [, { out_of_line_constraint attribute inline_constraint [inline_constraint]... }]]...</pre>
OID_clause	OBJECT IDENTIFIER IS { SYSTEM GENERATED PRIMARY KEY }
OID_index_clause	<pre>OIDINDEX [index] ({ physical_attributes_clause TABLESPACE tablespace } [physical_attributes_clause TABLESPACE tablespace]...)</pre>
on_comp_partitioned_table	<pre>[STORE IN (tablespace [, tablespace]...)] (PARTITION [partition [{ segment_attribute_clause key_compression</pre>

Subclause	Syntax
	<pre> } [segment_attribute_clause key_compression]...] [index_subpartition_clause]] [, PARTITION partition [{ segment_attribute_clause key_compression } [segment_attribute_clause key_compression]...] [index_subpartition_clause]]...]) </pre>
on_hash_partitioned_table	<pre> { STORE IN (tablespace[, tablespace]...) (PARTITION [partition [TABLESPACE tablespace]] [, PARTITION [partition [TABLESPACE tablespace]]]...) } </pre>
on_list_partitioned_table	<pre> (PARTITION partition [{ segment_attributes_clause key_compression } [segment_attributes_clause key_compression]...]) [, PARTITION partition [{ segment_attributes_clause key_compression } [segment_attributes_clause key_compression]...]]) ...) </pre>

Subclause	Syntax
on_object_clause	<pre> { schema.object { DIRECTORY directory_name JAVA { SOURCE RESOURCE } [schema.]object } } </pre>
on_range_partitioned_table	<pre> (PARTITION [partition [{ segment_attributes_clause key_compression } [segment_attributes_clause key_compression]...] [, PARTITION [partition [{ segment_attributes_clause key_compression } [segment_attributes_clause key_compression]...]]]...) </pre>
order_by_clause	<pre> ORDER [SIBLINGS] BY { expr position c_alias } [ASC DESC] [NULLS FIRST NULLS LAST] [, { expr position c_alias } [ASC DESC] [NULLS FIRST NULLS LAST]]... </pre>
out_of_line_constraint	<pre> [CONSTRAINT constraint_name] { UNIQUE (column [, column]...) PRIMARY KEY (column [, column]...) FOREIGN KEY (column [, column]...) references_clause CHECK (condition) } [constraint_state] </pre>
out_of_line_ref_constraint	<pre> { SCOPE FOR ({ ref_col ref_attr }) IS [schema.]scope_table REF ({ ref_col ref_attr }) WITH ROWID </pre>

Subclause	Syntax
	<pre> [CONSTRAINT constraint_name] FOREIGN KEY ({ ref_col ref_attr }) references_clause [constraint_state] } </pre>
outer_join_clause	<pre> table_reference [query_partition_clause] { outer_join_type JOIN NATURAL [outer_join_type] JOIN } table_reference [query_partition_clause] [ON condition USING (column [, column]...)] </pre>
outer_join_type	<pre> { FULL LEFT RIGHT } [OUTER] </pre>
parallel_clause	<pre> { NOPARALLEL PARALLEL [integer] } </pre>
parallel_enable_clause	<pre> PARALLEL_ENABLE [(PARTITION argument BY { ANY { HASH RANGE } (column [, column]...) }) [streaming_clause]] </pre>
partial_database_recovery	<pre> { TABLESPACE tablespace [, tablespace]... DATAFILE { 'filename' filenumber } [, 'filename' filenumber]... } STANDBY { TABLESPACE tablespace [, tablespace]... DATAFILE { 'filename' filenumber } [, 'filename' filenumber]... } } UNTIL [CONSISTENT WITH] CONTROLFILE } </pre>
partition_attributes	<pre> [{ physical_attributes_clause logging_clause allocate_extent_clause deallocate_unused_clause shrink_clause }] </pre>

Subclause	Syntax
	<pre> [physical_attributes_clause logging_clause allocate_extent_clause deallocate_unused_clause shrink_clause] ...] [OVERFLOW { physical_attributes_clause logging_clause allocate_extent_clause deallocate_unused_clause } [physical_attributes_clause logging_clause allocate_extent_clause deallocate_unused_clause] ...] [table_compression] [{ LOB LOB_item VARRAY varray } modify_LOB_parameters [{ LOB LOB_item VARRAY varray } modify_LOB_parameters] ...] </pre>
partition_extended_name	<pre> [schema.] { table view } [PARTITION (partition) SUBPARTITION (subpartition)] </pre>
partition_level_subpartition	<pre> { SUBPARTITIONS hash_subpartition_quantity [STORE IN (tablespace[, tablespace]...)] (subpartition_spec[, subpartition_spec]...) } </pre>
partition_spec	<pre> PARTITION [partition] [table_partition_description] </pre>
partitioning_storage_clause	<pre> [{ TABLESPACE tablespace OVERFLOW [TABLESPACE tablespace] LOB (LOB_item) STORE AS { LOB_segnane [(TABLESPACE tablespace)] (TABLESPACE tablespace) } VARRAY varray_item STORE AS LOB LOB_segnane } [{ TABLESPACE tablespace OVERFLOW [TABLESPACE tablespace] LOB (LOB_item) STORE AS { LOB_segnane [(TABLESPACE tablespace)] }]] </pre>

Subclause	Syntax
	<pre> (TABLESPACE tablespace) VARRAY varray_item STORE AS LOB LOB_segname] ...] </pre>
password_parameters	<pre> { { FAILED_LOGIN_ATTEMPTS PASSWORD_LIFE_TIME PASSWORD_REUSE_TIME PASSWORD_REUSE_MAX PASSWORD_LOCK_TIME PASSWORD_GRACE_TIME } { expr UNLIMITED DEFAULT } PASSWORD_VERIFY_FUNCTION { function NULL DEFAULT } } </pre>
permanent_tablespace_clause	<pre> { MINIMUM EXTENT integer [K M] BLOCKSIZE integer [K] logging_clause FORCE LOGGING DEFAULT [table_compression] storage_clause { ONLINE OFFLINE } extent_management_clause segment_management_clause flashback_mode_clause [MINIMUM EXTENT integer [K M] BLOCKSIZE integer [K] logging_clause FORCE LOGGING DEFAULT [table_compression] storage_clause { ONLINE OFFLINE } extent_management_clause segment_management_clause flashback_mode_clause] ... } </pre>
physical_attributes_clause	<pre> [{ PCTFREE integer PCTUSED integer INITTRANS integer storage_clause } [PCTFREE integer PCTUSED integer INITTRANS integer storage_clause]] ... </pre>

Subclause	Syntax
physical_properties	<pre> { segment_attributes_clause [table_compression] ORGANIZATION { HEAP [segment_attributes_clause] [table_compression] INDEX [segment_attributes_clause] index_org_table_clause EXTERNAL external_table_clause } CLUSTER cluster (column [, column]...) } </pre>
pragma_clause	<pre> PRAGMA RESTRICT_REFERENCES ({ method_name DEFAULT } , { RNDS WNDS RNPS WNPS TRUST } [, { RNDS WNDS RNPS WNPS TRUST }]...) </pre>
procedure_declaration	<pre> PROCEDURE name (parameter datatype [, parameter datatype]...) { IS AS } { pl/sql_block call_spec } </pre>
procedure_spec	<pre> PROCEDURE name (parameter datatype [, parameter datatype]...) [{ IS AS } call_spec] </pre>
proxy_authentication	<pre> { AUTHENTICATION REQUIRED AUTHENTICATED USING { PASSWORD DISTINGUISHED NAME CERTIFICATE [TYPE 'type'] [VERSION 'version'] } } </pre>
proxy_clause	<pre> { GRANT REVOKE } CONNECT THROUGH proxy [WITH { ROLE { role_name [, role_name]... ALL EXCEPT role_name [, role_name]... } NO ROLES }] [proxy_authentication] </pre>

Subclause	Syntax
qualified_disk_clause	<pre>search_string [NAME disk_name] [SIZE size_clause] [FORCE NOFORCE]</pre>
qualified_template_clause	<pre>template_name ATTRIBUTES ([MIRROR UNPROTECTED] [FINE COARSE])</pre>
query_partition_clause	<pre>PARTITION BY { value_expr[, value_expr]... (value_expr[, value_expr]...)}</pre>
query_table_expression	<pre>{ query_name [schema.] { table [{ PARTITION (partition) SUBPARTITION (subpartition) } [sample_clause] [sample_clause] @ dblink] { view materialized view } [@ dblink] } (subquery [subquery_restriction_clause]) table_collection_expression }</pre>
quiesce_clauses	QUIESCE RESTRICTED UNQUIESCE
range_partitioning	<pre>PARTITION BY RANGE (column[, column]...) (PARTITION [partition] range_values_clause table_partition_description [, PARTITION [partition] range_values_clause table_partition_description]...)</pre>
range_values_clause	<pre>VALUES LESS THAN ({ value MAXVALUE } [, { value MAXVALUE }]...)</pre>
rebalance_diskgroup_clause	REBALANCE [POWER integer]

Subclause	Syntax
rebuild_clause	<pre> REBUILD [{ PARTITION partition SUBPARTITION subpartition } { REVERSE NOREVERSE }] [parallel_clause TABLESPACE tablespace PARAMETERS ('ODCI_parameters') ONLINE COMPUTE STATISTICS physical_attributes_clause key_compression logging_clause] [parallel_clause TABLESPACE tablespace PARAMETERS ('ODCI_parameters') ONLINE COMPUTE STATISTICS physical_attributes_clause key_compression logging_clause] ... </pre>
records_per_block_clause	{ MINIMIZE NOMINIMIZE } RECORDS_PER_BLOCK
recover_clause	<pre> { { DISCONNECT [FROM SESSION] { TIMEOUT integer NOTIMEOUT } } { NODELAY DEFAULT DELAY DELAY integer } NEXT integer { EXPIRE integer NO EXPIRE } parallel_clause USING CURRENT LOGFILE UNTIL CHANGE integer THROUGH { [THREAD integer] SEQUENCE integer ALL ARCHIVELOG { ALL LAST NEXT } SWITCHOVER } } [{ DISCONNECT [FROM SESSION] { TIMEOUT integer NOTIMEOUT } } { NODELAY DEFAULT DELAY DELAY integer } NEXT integer { EXPIRE integer NO EXPIRE } parallel_clause USING CURRENT LOGFILE UNTIL CHANGE integer THROUGH { [THREAD integer] SEQUENCE integer ALL ARCHIVELOG } </pre>

Subclause	Syntax
	<pre> { ALL LAST NEXT } SWITCHOVER }] ... </pre>
recovery_clauses	<pre> { general_recovery managed_standby_recovery BEGIN BACKUP END BACKUP } </pre>
redo_log_file_spec	<pre> ['filename' ('filename' [, 'filename'] ...)] [SIZE size_clause] [REUSE] </pre>
redo_thread_clauses	<pre> { ENABLE DISABLE } { INSTANCE 'instance_name' [PUBLIC] THREAD integer } </pre>
reference_model	<pre> REFERENCE reference_spreadsheet_name ON (subquery) spreadsheet_column_clauses [cell_reference_options] </pre>
references_clause	<pre> REFERENCES [schema.] { object_table view } [(column [, column] ...)] [ON DELETE { CASCADE SET NULL }] [constraint_state] </pre>
referencing_clause	<pre> REFERENCING { OLD [AS] old NEW [AS] new PARENT [AS] parent } [OLD [AS] old NEW [AS] new PARENT [AS] parent]... </pre>
register_logfile_clause	<pre> REGISTER [OR REPLACE] [PHYSICAL LOGICAL] LOGFILE [file_specification [, file_specification] ...] FOR logminer_session_name </pre>

Subclause	Syntax
relational_properties	<pre> { column datatype [SORT] [DEFAULT expr] [inline_constraint [inline_constraint]... inline_ref_constraint] { out_of_line_constraint out_of_line_ref_constraint supplemental_logging_props } } [, { column datatype [SORT] [DEFAULT expr] [inline_constraint [inline_constraint]... inline_ref_constraint] { out_of_line_constraint out_of_line_ref_constraint supplemental_logging_props }]]... </pre>
relational_table	<pre> CREATE [GLOBAL TEMPORARY] TABLE [schema.]]table [(relational_properties)] [ON COMMIT { DELETE PRESERVE } ROWS] [physical_properties] [table_properties] ; </pre>
rename_column_clause	RENAME COLUMN old_name TO new_name
rename_index_partition	<pre> RENAME { PARTITION partition SUBPARTITION subpartition } TO new_name </pre>
rename_partition_subpart	<pre> RENAME { PARTITION SUBPARTITION } current_name TO new_name </pre>
replace_type_clause	<pre> REPLACE [invoker_rights_clause] AS OBJECT (attribute datatype [, attribute datatype]... [, element_spec [, element_spec]...]) </pre>
resize_disk_clauses	<pre> RESIZE { ALL [SIZE size_clause] DISK disk_name [SIZE size_clause] [, disk_name [SIZE size_clause]]... DISKS IN FAILGROUP </pre>

Subclause	Syntax
	<pre> failgroup_name [SIZE size_clause] [, failgroup_name [SIZE size_clause]]... } </pre>
resource_parameters	<pre> { { SESSIONS_PER_USER CPU_PER_SESSION CPU_PER_CALL CONNECT_TIME IDLE_TIME LOGICAL_READS_PER_SESSION LOGICAL_READS_PER_CALL COMPOSITE_LIMIT } { integer UNLIMITED DEFAULT } PRIVATE_SGA { integer [K M] UNLIMITED DEFAULT } } </pre>
restricted_session_clauses	{ ENABLE DISABLE } RESTRICTED SESSION
return_clause	<pre> { RETURN datatype [{ IS AS } call_spec] sqlj_object_type_sig } </pre>
return_rows_clause	RETURN { UPDATED ALL } ROWS
returning_clause	RETURNING expr [, expr]... INTO data_item [, data_item]...
revoke_object_privileges	<pre> { object_privilege ALL [PRIVILEGES] } [, { object_privilege ALL [PRIVILEGES] }] ... on_object_clause FROM grantee_clause [CASCADE CONSTRAINTS FORCE] </pre>
revoke_system_privileges	<pre> { system_privilege role ALL PRIVILEGES } [, { system_privilege role ALL PRIVILEGES }] ... FROM grantee_clause </pre>
rollup_cube_clause	{ ROLLUP CUBE } (grouping_expression_list)

Subclause	Syntax
routine_clause	[schema.] [type. package.] { function procedure method } [@dblink_name] ([argument [, argument]...])
row_movement_clause	{ ENABLE DISABLE } ROW MOVEMENT
sample_clause	SAMPLE [BLOCK] (sample_percent) [SEED (seed_value)]
schema_object_clause	{ object_option [, object_option]... ALL } auditing_on_clause
scoped_table_ref_constraint	{ SCOPE FOR ({ ref_column ref_attribute }) IS [schema.] { scope_table_name c_alias } } , SCOPE FOR ({ ref_column ref_attribute }) IS [schema.] { scope_table_name c_alias } }]...
searched_case_expression	WHEN condition THEN return_expr [WHEN condition THEN return_expr]...
security_clause	GUARD { ALL STANDBY NONE }
segment_attributes_clause	{ physical_attributes_clause TABLESPACE tablespace logging_clause } [physical_attributes_clause TABLESPACE tablespace logging_clause]...
segment_management_clause	SEGMENT SPACE MANAGEMENT { MANUAL AUTO }
select_list	{ * { query_name.* [schema.] { table view materialized view } .* expr [[AS] c_alias] } , { query_name.* [schema.] { table view materialized view } .*

Subclause	Syntax
	<pre> expr [[AS] c_alias] }] ... } </pre>
set_subpartition_template	<pre> SET SUBPARTITION TEMPLATE { (SUBPARTITION subpartition [list_values_clause] [partitioning_storage_clause] [, SUBPARTITION subpartition [list_values_clause] [partitioning_storage_clause]...]) hash_subpartition_quantity } </pre>
set_time_zone_clause	<pre> SET TIME_ZONE = '{ { + - } hh : mi time_zone_region }' </pre>
shrink_clause	SHRINK SPACE [COMPACT] [CASCADE]
shutdown_dispatcher_clause	SHUTDOWN [IMMEDIATE] dispatcher_name
simple_case_expression	<pre> expr WHEN comparison_expr THEN return_expr [WHEN comparison_expr THEN return_expr]... </pre>
single_column_for_loop	<pre> FOR dimension_column { IN ({ literal [, literal]... subquery }) [LIKE pattern] FROM literal TO literal { INCREMENT DECREMENT } literal } </pre>
single_table_insert	<pre> insert_into_clause { values_clause [returning_clause] subquery } </pre>
size_clause	integer [K M G T]

Subclause	Syntax
split_index_partition	<pre>SPLIT PARTITION partition_name_old AT (value [, value]...) [INTO (index_partition_description, index_partition_description)] [parallel_clause]</pre>
split_table_partition	<pre>SPLIT PARTITION current_partition { AT VALUES } (value [, value]...) [INTO (partition_spec, partition_spec)] [update_index_clauses] [parallel_clause]</pre>
split_table_subpartition	<pre>SPLIT SUBPARTITION subpartition VALUES ({ value NULL } [, value NULL]...) [INTO (subpartition_spec, subpartition_spec)] [update_index_clauses] [parallel_clause]</pre>
sql_statement_clause	<pre>{ { statement_option ALL } [, { statement_option ALL }]... { system_privilege ALL PRIVILEGES } [, { system_privilege ALL PRIVILEGES }]... } [auditing_by_clause]</pre>
sqlj_object_type	<pre>EXTERNAL NAME java_ext_name LANGUAGE JAVA USING (SQLData CustomDatum OraData)</pre>
sqlj_object_type_attr	<pre>EXTERNAL NAME 'field_name'</pre>
sqlj_object_type_sig	<pre>RETURN { datatype SELF AS RESULT } EXTERNAL { VARIABLE NAME 'java_static_field_name' NAME 'java_method_sig' }</pre>
standby_database_clauses	<pre>(activate_standby_db_clause maximize_standby_db_clause register_logfile_clause commit_switchover_clause start_standby_clause stop_standby_clause) [parallel_clause]</pre>

Subclause	Syntax
start_standby_clause	<pre>START LOGICAL STANDBY APPLY [IMMEDIATE] [NODELAY] [NEW PRIMARY dblink INITIAL [scn_value] { SKIP FAILED TRANSACTION FINISH }]</pre>
startup_clauses	<pre>{ MOUNT [{ STANDBY CLONE } DATABASE] OPEN { [READ WRITE] [RESETLOGS NORESETLOGS] [UPGRADE DOWNGRADE] READ ONLY } }</pre>
stop_standby_clause	<pre>{ STOP ABORT } LOGICAL STANDBY APPLY</pre>
storage_clause	<pre>STORAGE ({ INITIAL integer [K M] NEXT integer [K M] MINEXTENTS integer MAXEXTENTS { integer UNLIMITED } PCTINCREASE integer FREELISTS integer FREELIST GROUPS integer OPTIMAL [integer [K M] NULL] BUFFER_POOL { KEEP RECYCLE DEFAULT } } [INITIAL integer [K M] NEXT integer [K M] MINEXTENTS integer MAXEXTENTS { integer UNLIMITED } PCTINCREASE integer FREELISTS integer FREELIST GROUPS integer OPTIMAL [integer [K M] NULL] BUFFER_POOL { KEEP RECYCLE DEFAULT }] ...)</pre>
streaming_clause	<pre>{ ORDER CLUSTER } BY (column [, column]...)</pre>
subpartition_by_hash	<pre>SUBPARTITION BY HASH (column [, column]...) [SUBPARTITIONS quantity STORE IN (tablespace [, tablespace]...)]</pre>

Subclause	Syntax
	<pre> subpartition_template]</pre>
subpartition_by_list	SUBPARTITION BY LIST (column) [subpartition_template]
subpartition_spec	SUBPARTITION [subpartition] [list_values_clause] [partitioning_storage_clause]
subpartition_template	SUBPARTITION TEMPLATE (SUBPARTITION subpartition [list_values_clause] [partitioning_storage_clause] [, SUBPARTITION subpartition [list_values_clause] [partitioning_storage_clause]]) hash_subpartition_quantity
subprogram_declaration	{ MEMBER STATIC } { procedure_declaration function_declaration constructor_declaration }
subprogram_spec	{ MEMBER STATIC } { procedure_spec function_spec }
subquery	[subquery_factoring_clause] SELECT [hint] [{ { DISTINCT UNIQUE } ALL }] select_list FROM table_reference [, table_reference]... [where_clause] [hierarchical_query_clause] [group_by_clause] [HAVING condition] [model_clause] [{ UNION [ALL INTERSECT MINUS }] (subquery)

Subclause	Syntax
] [order_by_clause]
subquery_factoring_clause	WITH query_name AS (subquery) [, query_name AS (subquery)]...
subquery_restriction_clause	WITH { READ ONLY CHECK OPTION [CONSTRAINT constraint] }
substitutable_column_clause	[ELEMENT] IS OF [TYPE] ([ONLY] type) [NOT] SUBSTITUTABLE AT ALL LEVELS
supplemental_db_logging	{ ADD DROP } SUPPLEMENTAL LOG { DATA supplemental_id_key_clause }
supplemental_id_key_clause	DATA ({ ALL PRIMARY KEY UNIQUE FOREIGN KEY } [, { ALL PRIMARY KEY UNIQUE FOREIGN KEY }]...) COLUMNS
supplemental_log_grp_clause	GROUP log_group (column [NO LOG] [, column [NO LOG]]...) [ALWAYS]
supplemental_logging_props	{ supplemental_log_grp_clause supplemental_id_key_clause }
supplemental_table_logging	{ ADD SUPPLEMENTAL LOG { supplemental_log_grp_clause supplemental_id_key_clause } [, SUPPLEMENTAL LOG { supplemental_log_grp_clause supplemental_id_key_clause }}

Subclause	Syntax
	<pre>]... DROP SUPPLEMENTAL LOG { supplemental_id_key_clause GROUP log_group } [, SUPPLEMENTAL LOG { supplemental_id_key_clause GROUP log_group }]... } </pre>
table_collection_expression	TABLE (collection_expression) [(+)]
table_compression	{ COMPRESS NOCOMPRESS }
table_index_clause	<pre> [schema.]table [t_alias] (index_expr [ASC DESC] [, index_expr [ASC DESC]]...) [index_properties] </pre>
table_partition_description	<pre> [segment_attributes_clause] [table_compression key_compression] [OVERFLOW [segment_attributes_clause]] [{ LOB_storage_clause varray_col_properties } [LOB_storage_clause varray_col_properties]...] [partition_level_subpartition] </pre>
table_partitioning_clauses	<pre> { range_partitioning hash_partitioning list_partitioning composite_partitioning } </pre>
table_properties	<pre> [column_properties] [table_partitioning_clauses] [CACHE NOCACHE] [parallel_clause] [ROWDEPENDENCIES NOROWDEPENDENCIES] [enable_disable_clause] [enable_disable_clause]... [row_movement_clause] [AS subquery] </pre>

Subclause	Syntax
table_reference	<pre> { ONLY (query_table_expression) [flashback_query_clause] [t_alias] query_table_expression [flashback_query_clause] [t_alias] (join_clause) join_clause } </pre>
tablespace_clauses	<pre> { EXTENT MANAGEMENT LOCAL DATAFILE file_specification [, file_specification]... SYSAUX DATAFILE file_specification [, file_specification]... default_tablespace default_temp_tablespace undo_tablespace } </pre>
tablespace_group_clause	TABLESPACE GROUP { tablespace_group_name '' }
tablespace_logging_clauses	<pre> { logging_clause [NO] FORCE LOGGING } </pre>
tablespace_retention_clause	RETENTION { GUARANTEE NOGUARANTEE }
tablespace_state_clauses	<pre> { ONLINE OFFLINE [NORMAL TEMPORARY IMMEDIATE] } READ { ONLY WRITE } { PERMANENT TEMPORARY } </pre>
temporary_tablespace_clause	<pre> TEMPORARY TABLESPACE tablespace [TEMPFILE file_specification [, file_specification]...] [tablespace_group_clause] [extent_management_clause] </pre>
text	<pre> [N n] { 'c [c]...' { Q q } 'quote_delimiter c [c]... quote_delimiter' } </pre>

Subclause	Syntax
trace_file_clause	TRACE [AS 'filename' [REUSE]] [RESETLOGS NORESETLOGS]
truncate_partition_subpart	TRUNCATE { PARTITION partition SUBPARTITION subpartition } [{ DROP REUSE } STORAGE] [update_index_clauses [parallel_clause]]
undo_tablespace	[BIGFILE SMALLFILE] UNDO TABLESPACE tablespace [TABLESPACE file_specification [, file_specification]...]
undo_tablespace_clause	UNDO TABLESPACE tablespace [DATAFILE file_specification [, file_specification]...] [extent_management_clause] [tablespace_retention_clause]
undrop_disk_clause	UNDROP DISKS
update_all_indexes_clause	UPDATE INDEXES [(index ({ update_index_partition update_index_subpartition })) [, (index ({ update_index_partition update_index_subpartition }))]...]
update_global_index_clause	{ UPDATE INVALIDATE } GLOBAL INDEXES
update_index_clauses	{ update_global_index_clause update_all_indexes_clause }
update_index_partition	PARTITION [partition] [index_partition_description [index_subpartition_clause]] [, PARTITION [partition] [index_partition_description

Subclause	Syntax
	<pre> [index_subpartition_clause]]]... </pre>
update_index_subpartition	<pre> SUBPARTITION [subpartition] [TABLESPACE tablespace] [, SUBPARTITION [subpartition] [TABLESPACE tablespace]]... </pre>
update_set_clause	<pre> SET { { (column [, column]...) = (subquery) column = { expr (subquery) DEFAULT } } [, { (column [, column]...) = (subquery) column = { expr (subquery) DEFAULT } }]]... VALUE (t_alias) = { expr (subquery) } } </pre>
upgrade_table_clause	<pre> UPGRADE [[NOT] INCLUDING DATA] [column_properties] </pre>
using_function_clause	<pre> USING [schema.] [package. type.] function_name </pre>
using_index_clause	<pre> USING INDEX { [schema.] index (create_index_statement) index_properties } </pre>
using_statistics_type	<pre> USING { [schema.] statistics_type NULL } </pre>
using_type_clause	<pre> USING [schema.] implementation_type [array_DML_clause] </pre>
validation_clauses	<pre> { VALIDATE REF UPDATE [SET DANTLING TO NULL] VALIDATE STRUCTURE [CASCADE] [into_clause] { OFFLINE ONLINE } } </pre>

Subclause	Syntax
values_clause	VALUES ({ expr DEFAULT } [, { expr DEFAULT }]...)
varray_col_properties	VARRAY varray_item { [substitutable_column_clause] STORE AS LOB { [LOB_segname] (LOB_parameters) LOB_segname } substitutable_column_clause }
where_clause	WHERE condition
windowing_clause	{ ROWS RANGE } { BETWEEN { UNBOUNDED PRECEDING CURRENT ROW value_expr { PRECEDING FOLLOWING } } AND { UNBOUNDED FOLLOWING CURRENT ROW value_expr { PRECEDING FOLLOWING } } { UNBOUNDED PRECEDING CURRENT ROW value_expr PRECEDING } }
XML_attributes_clause	XMLATTRIBUTES (value_expr [AS c_alias] [, value_expr [AS c_alias]]...)
XMLSchema_spec	[XMLSCHEMA XMLSchema_URL] ELEMENT { element XMLSchema_URL # element }
XMLType_column_properties	XMLTYPE [COLUMN] column [XMLType_storage] [XMLSchema_spec]
XMLType_storage	STORE AS { OBJECT RELATIONAL CLOB { [LOB_segname [(LOB_parameters)] LOB_parameters } }

Subclause	Syntax
XMLType_table	<pre>CREATE TABLE [GLOBAL TEMPORARY] TABLE [schema.]table OF XMLTYPE [(object_properties)] [XMLTYPE XMLType_storage] [XMLSchema_spec] [ON COMMIT { DELETE PRESERVE } ROWS] [OID_clause] [OID_index_clause] [physical_properties] [table_properties] ;</pre>
XMLType_view_clause	<pre>OF XMLTYPE [XMLSchema_spec] WITH OBJECT IDENTIFIER { DEFAULT (expr [, expr]...) }</pre>

Appendix D

Oracle Background Processes

Oracle Background Processes

This appendix is not an exhaustive list of all background processes and is meant to serve as a quick reference. For more details on these background processes and any that have not been mentioned here, consult the Oracle Database Reference guide.

General Processes

Acronym	Process Name	Description	Required for basic operation	Started by default
ARC n	Archiver Process	Writes filled redo logs to the archive log location(s). Possible processes include ARC0–ARC9 and ARCa–ARCb.	No	No
CJQ0	Job Queue Coordinator Process	Spawns slave processes (Jnnn) to execute jobs in the queue.	No	Yes
CKPT	Checkpoint Process	Writes checkpoint information to control files and data file headers.	Yes	Yes
DBRM	Database Resource Manager Process	Sets resource plans and performs other Resource Manager tasks.	No	Yes
DBW n	Database Writer Process	Writes dirty buffers from the buffer cache to the data files. Possible processes include DBW0–DBW9 and DBWa–DBWz.	Yes	Yes
DIA0	Diagnostic Process	Responsible for hang detection and deadlock resolution. Triggers DIAG to perform diagnostic tasks.	Yes	Yes
DIAG	Diagnostic Capture Process	Performs diagnostic dumps and executes global oradebug commands	Yes	Yes
Dnnn	Dispatcher Process	In a Shared Server configuration, dispatchers place connection requests in a connection request queue.	No	Yes
EMNC	EMON Coordinator Process	Coordinates the even management and notification activity in the database, including Streams Event Notifications, Continuous Query Notifications, and Fast Application Notifications.	No	No
FBDA	Flashback Data Archiver Process	Archives historical rows for tracked tables into flashback data archives and manages the flashback archives.	No	Yes
FMON	File Mapping Monitor Process	Spawns FMPUTL, an external non-Oracle Database process that communicates with the mapping libraries provided by storage vendors. Responsible for managing the mapping information.	No	No
Jnnn	Job Queue Slave Process	Processes jobs in the queue. Spawns by CJQ0.	No	Yes
LGWR	Log Writer Process	Writes the log buffer out to the redo logs.	Yes	Yes
MMAN	Memory Manager Process	Serves as the SGA Memory Broker and coordinates the sizing of the memory components.	No	Yes
MMNL	Manageability Monitor Lite Process	Performs frequent and lightweight manageability-related tasks, such as session history, capture and metrics computation.	No	Yes
MMON	Manageability Monitor Process	Collects statistics for the Automatic Workload Repository.	No	Yes
PMON	Process Monitor	Recover failed process resources. If Shared Server architecture is used, PMON monitors and	Yes	Yes

Acronym	Process Name	Description	Required for basic operation	Started by default
		restarts any failed dispatcher or server processes.		
Pnnn	Parallel Query Slave Process	Started and stopped as needed to participate in parallel query operations.	No	No
PSP0	Process Spawner Process	Starts and stops Oracle processes. Reduces workload of RBAL by starting/stopping ASM rebalance slaves.	No	Yes
RCBG	Result Cache Background Process	Supports SQL query and PLO/SQL function result caches.	No	No
RECO	Recoverer Process	Resolves failures involving distributed transactions.	No	Yes
RVWR	Recovery Writer Process	Writes flashback data to flashback database logs in the flash recovery area.	No	No
SMCO	Space Management Coordinator Process	Coordinates the execution of various space management related tasks, such as proactive space allocation and space reclamation.	No	Yes
SMON	System Monitor Process	Performs recovery after instance failure and monitors temporary segments and extents.	Yes	Yes
Snnn	Shared Server Process	In a shared Server configuration, shared servers check a connection request queue (populated by dispatchers) and services the connection requests.	No	Yes
VKTM	Virtual Keeper of Time Process	Responsible for providing a wall-clock time (updated every second) and reference-tin counter (updated every 20ms and available only when running at elevated priority).	Yes	Yes
Wnnn	Space Management Slave Process	These are slave processes spawned by SMCO to execute space management tasks.	No	Yes

Advanced Queuing Processes

Acronym	Process Name	Description	Required for basic operation	Started by default
Qnnn	AQ Server Class Process	Processes messaged in the Streams AQ queue. Spawned by QMNC.	No	Yes
QMNC	AQ Coordinator Process	Monitors message queues. Spawns Qnnn processes.	No	Yes

Automatic Storage Management Processes

Acronym	Process Name	Description	Required for basic operation	Started by default
ARBn	ASM Rebalance Process	Rebalances data extents across the ASM file systems. Possible processes are ARB0 – ARB9 and ARBA.	No	No
ASMB	ASM Background Process	Communicates with the Automatic Storage Management instance.	No	No
DSKM	Slave Diskmon Process	Acts as conduit between RDBMS and ASM instances and the Master Diskmon daemon to communicate I/O Fencing Information, I/O Resource Manager Plans, and Transaction Commit Cache information to SAGE storage. If no SAGE storage is used, the slave diskmon	No	No

Acronym	Process Name	Description	Required for basic operation	Started by default
		process will exit silently after startup of the instance.		
GMON	ASM Disk Group Monitor Process	Maintains disk membership in ASM disk groups.	No	No
MARK	Mark AU for Resynchronization Coordinator Process	Marks ASM Allocation Units as stale following a missed write to an offline disk.	No	No
RBAL	ASM Rebalance Master Process	Coordinates rebalance activity for disk groups in an Automatic Storage Management instance. Performs global opens on Automatic Storage Management disks.	No	No
DRnn	ASM Disk Resynchronization Slave Process	Resynchronizes the contents of an offline disk. When a disk online SQL command is issued on a disk or disks that are offline, ASM spawns DRnn. Depending on the load, more than one slave may be spawned.	No	No

Data Guard Processes

Acronym	Process Name	Description	Required for basic operation	Started by default
DMON	Data Guard Broker Monitor Process	Starts when Data Guard Broker is started. DMON is the server-side component that interacts with the local database and the DMON processes of the other databases to perform the requested function. Also responsible for monitoring the health of the broker configuration and for ensuring that every database has a consistent description of the configuration.	No	No
FSFP	Data Guard Broker Fast Start Failover Pinger Process	Observer process integrated in the DGMGROL client-side component. It monitors the DG Broker environment and initiates the failover on detecting a problem with Primary.	No	No
INSV	Data Guard Broker Instance Slave Process	Performs Data Guard Broker communication between instances of a RAC database.	No	No
LSP0	Logical Standby Coordinator Process	Assigns transactions to different Data Guard appliers and coordinates among them to ensure that dependencies between transactions are honored.	No	No
LSP1	Logical Standby Dictionary Build Process	Builds the data dictionary for the Logical Standby.	No	No
LSP2	Logical Standby Set Guard Process	Maintains Guard Standby information for the Logical Standby.	No	No
MRP0	Managed Standby Recovery Process	Provides transparent support for XA global transactions in RAC.	No	No
NSVn	Data Guard Broker NetSlave Process	Performs broker network communications between databases in Data Guard configuration.	No	No

Acronym	Process Name	Description	Required for basic operation	Started by default
		Possible processes include NSV0–NSV9.		
RSM0	Data Guard Broker Worker Process	Used by the DMON process to manage and monitor the database.	No	No

RMAN Processes

Acronym	Process Name	Description	Required for basic operation	Started by default
CTWR	Change Tracking Writer Process	Writes to the RMAN Change Tracking Log, a bitmap representing the entire database. The bitmap has an associated SCN, which is the SCN as at the last backup.	No	No

Real Application Clusters Processes

Acronym	Process Name	Description	Required for basic operation	Started by default
ACMS	Atomic Control File to Memory Server Process	Contributes to ensuring a distributed SGA memory update is either globally omitted on success or globally aborted in the event of a failure in an Oracle RAC environment.	No	No
GTXn	Global Transaction Process	Provides transparent support for XA global transactions in an Oracle RAC environment. The database auto tunes the number of these process based on the workload of XA global transactions. Possible processes are GTX0–GTX9 and GTXa–GTXj.	No	No
LCK0	Instance Enqueue Background Process	Manage the global enqueued requests and the cross-instance broadcast. Handles all requests for resources other than data blocks.	No	No
LMD0	Global Enqueue Service Daemon 0 Process	Manages enqueue manager service requests for Global Cache Service enqueues to control access to global enqueues and resources. The LMD0 process also handles deadlock detection and remote enqueued requests. Remote resource requests are the requests originating from another instance.	No	No
LMON	Global Enqueue Service Monitor Process	Monitors the entire cluster to manage the global enqueues and the resources. Manages instance and process failures and the associated recovery for the Global Cache Service (GCS) and Global Enqueue Service (GES). In particular LMON handles the part of recovery associated with global resources. LMON-provided services are also known as cluster group services (CGS).	No	No
LMSn	Global Cache Service Process	Handles remote Global Cache Service (GCS) messages. The number of LMS processes varies depending on the amount of messaging traffic among nodes in the cluster.	No	No
PING	Interconnect Latency Measurement Process	Assesses the latencies associated with communications for each pair of instances. Every	No	No

Acronym	Process Name	Description	Required for basic operation	Started by default
		few seconds, the process in one instance (INSTANCE_NUMBER value) sends two messages to each instance (TARGET_INSTANCE value). One message has a size of 500 bytes and the other has a size of 8 KB. The message is received by the PING process on the target instance and is immediately acknowledged. The time for the round-trip is measured and collected.		
RMSn	Oracle RAC Management Process	Performs manageability tasks for Oracle RAC, such as creation of RAC-related resources when new instances are added to the clusters.	No	No
RSMN	Remote Slave Monitor Process	Manages background slave process creation and communication on remote instances.	No	No

Appendix E

Acronyms and

Terms

Term	Definition
Active session pool	Number of current active sessions allowed for a resource group or subplan
ADDM	Automatic Database Diagnostic Monitor
ASM	Automatic Storage Management
ASM	Automatic Summary Management
ASMM	Automatic Shared Memory Management
ASSM	Automatic Segment Space Management
ATO	Automatic Tuning Optimizer
Automatic PGA Memory Management	A feature of the Oracle database that simplifies and improves the way PGA memory is allocated
Automatic Shared Memory Management	A feature of the Oracle database that automates the management of the most important shared memory structures used by an Oracle database instance
Automatic Storage Management	A mechanism that provides a vertical integration of the file system and the volume manager, specifically built for the Oracle database files
Automatic Database Diagnostic Management	A utility that performs a top-down instance analysis, identifies problems and potential causes, and makes recommendations for fixing the problems
Automatic Tuning Optimizer	A database feature that performs various analyses of SQL performance within the database
Automatic Workload Repository	An infrastructure that collects, processes, and maintains performance statistics for problem detection and self-tuning purposes
Auxiliary database	A database that is used when creating a duplicate database or performing tablespace point-in-time recovery
AWR	Automatic Workload Repository
Backup piece	An individual file that is part of a backup set
Backup set	A copy of one or more data or archived log files. It differs from image copies in that empty blocks are not stored.
Block change tracking	A feature that uses the change tracking writer (CTWR) background process to record the physical location of all database changes in a separate file
Block corruption	Corruption of a data block. A corrupted data block is a block that is not in a recognized Oracle format, or whose contents are not internally consistent.
Block Media Recovery	A recovery method that reduces the smallest recoverable unit of media recovery from a data file to a block
BMR	Block Media Recovery
Buffer cache	A region of memory that caches blocks of data retrieved from the database
CFS	Cluster File Storage

Term	Definition
Change tracking file	A file used to store the physical location of database changes made since the last backup
Channel	A link or connection to a target database
CLI	Command-line interpreter
Cluster	A group of one or more tables that share the same data blocks
CMAN	Oracle Connection Manager. It functions as a net traffic firewall and proxy server.
Control file	A file that contains information about the physical structure of the database, including the locations of all data and redo log files
CRS	Cluster Ready Services
Data block	The smallest unit of physical storage within the database. Data blocks contain rows of data, index information, and so on.
Data dictionary cache	An area of memory within the shared pool that holds the definitions of dictionary objects in memory
Data file	A file that contains data for the database
Database Character Set Scanner	A utility that assesses the feasibility of migrating an Oracle database to a new database character set
DBA	Database administrator
DBA	Data block address, which is used to uniquely identify a data block within the database
DBCA	Database Configuration Assistant
DBVERIFY	An external command-line utility that performs a physical data structure integrity check on an offline database
DDL	Data definition language. It is the class of SQL statements that define and manipulate database objects.
DML	Data manipulation language. It is the class of SQL statements that query and manipulate data.
EM	Enterprise Manager
emctl	Enterprise Manager Control. It is a utility for starting, stopping, and checking the status of Database Control, the Oracle Agent, and Oracle Management servers.
Encoded character set	A character set that maps numeric codes to characters that a computer or terminal can display and receive
Enterprise Manager Database Control Console	A graphical interface used to manage the database
EXTPROC	External code libraries
Fast recovery area	A unified storage location for all recovery-related files and activities in an Oracle database
FGA	Fine-grained auditing
FGAC	Fine-Grained Access Control
Flash recovery area	Renamed to Fast recovery area.
Flashback buffer	An area in memory that stores Flashback Database data
Flashback Database	A new recovery method that uses Undo data, instead of Redo data, to recover the database

Term	Definition
Flashback Drop	A feature that enables you to undo the effects of a <code>DROP TABLE</code> statement without resorting to traditional point-in-time recovery
Flashback Table	A command that enables you to recover a table and all its dependent objects from the recycle bin
Flashback Transaction Query	A diagnostic tool that you can use to view changes made to the database at the transaction level
Flashback Versions Query	A query syntax that provides a history of changes made to a row along with the corresponding identifier of the transaction that made the change
Format mask elements	A character literal that describes the format of datetime or numeric data stored in a character string
Growth trend report	Analysis of the growth of database segments
Globalization support	A feature set that ensures that utilities and error messages, sort order, alphabet, calendar, date, time, money, and numbers automatically adapt to the native language
Image copy	A bit-for-bit identical copy of a database file
Incarnation	A separate version of a physical database. The incarnation of the database changes when you open it with the <code>RESETLOGS</code> option, but you can recover backups from an earlier incarnation so long as the necessary redo is available.
Index-organized tables	A database structure that has the appearance of a table but stores its data in a B*Tree structure
<code>init.ora</code> or <code>init<sid>.ora</code>	The initialization parameter file that controls how the database instance is configured and run at startup time. Also known as “parameter file”.
Instance	The collection of shared memory and processes used to access the Oracle database
IPC	Internal Process Communication
<code>isqlplusctl</code>	Control utility for starting and stopping <i>i</i> SQL*Plus listener processes
ISV	Independent software vendor
Java pool	A region of memory in the SGA that is used for all session-specific Java code and data within the Java Virtual Machine (JVM)
JDBC	Java Database Connectivity
<code>jnnn</code>	Job Queue Processes. They execute scheduled jobs.
Keep buffer cache	An area of memory in the SGA used to cache data in the buffer cache for longer periods of time
Language and Character Set File Scanner	A statistic-based utility for determining the language and character set for unknown file text
Large pool	An optional memory storage area used for buffering large I/O requests

Term	Definition
LCSSCAN	Language and Character Set File Scanner
LEGATO® NetWorker, Single-Server Version	Software included with Oracle Database 10g that enables the Recovery Manager utility to write to tape drives
Library cache	An area of memory within the shared pool that contains the fully parsed or compiled representations of PL/SQL blocks and SQL statements
Linguistic sort	A feature that produces a sort sequence that matches the alphabetic sequence of characters, and not their numeric values in the character encoding scheme
Listener	The gateway to the Oracle instance for all nonlocal user connections
Locale	A collection of information about the linguistic and cultural preferences from a particular region
Locale variants	A language-dependent territory definition
LSSV	LEGATO® NetWorker, Single-Server Version
Media management library	An interface used by RMAN when writing to or reading from tapes
Memory Advisor	A feature of Enterprise Manager that helps you tune the size of your memory structures
Memory Manager (MMAN)	A database background process that serves as the SGA memory broker and coordinates the sizing of memory components
Metric	A measurement of some database or instance characteristic
MML	Media management library
MMON	Management Monitor Process. This process issues alerts whenever a metric violates its threshold value. It captures statistics for SQL objects that have been recently modified.
National Language Support	Parameters and files that determine the locale-specific behavior of the database client and the database server
nK block size buffer	A region of memory in the SGA, which caches data blocks that are of a different size than the default database block size. It is used to support transportable tablespaces.
NLS	National Language Support
NLS Runtime Library	A comprehensive suite of language-independent functions that allow proper text and character processing and language-convention manipulations
NLS_LANG	An environment variable used to specify the language, territory, and character set used by a database
NLSRTL	National Language Support Runtime Library
NMP	Named Pipes
OC4J	Oracle Application Server Containers for J2EE
OMF	Oracle Managed Files
Optimizer statistics	Statistics that describe the database and the objects in the database, and are used by the query optimizer to choose the best execution plan for each SQL statement

Term	Definition
OUI	Oracle Universal Installer
Oracle Locale Builder	A feature that provides a graphical user interface through which you can easily view, modify, and define locale-specific data
Oracle Managed Files	A feature of the Oracle database, which manages the creation, naming, and deletion of Oracle database files within dedicated areas of the disk
Oracle Net	An interface that enables network connections between Oracle Database 10g and client or middle-tier applications
Oracle Shared Server	A database server configuration that allows many user processes to share a small number of server processes, minimizing the number of server processes and maximizing the use of available system resources
ORACLE_BASE	Environment variable used to point to the base of the OFA structure
ORACLE_HOME	Environment variable used to identify a directory containing Oracle software
ORACLE_SID	Environment variable used to specify the default database instance name
Package	A collection of procedures and function definitions that are logically related. The procedures and functions are implemented by the package body.
Parallelization	Allocation of multiple channels for RMAN backup and recovery operations
PGA	Program Global Area
PGA Advisor	A feature of Enterprise Manager that gives detailed statistics for the work areas and provides recommendations about optimal usage of Program Global Area (PGA) memory on the basis of workload characteristics
Pipe	An area of memory used by one process to pass information to another
Private SQL area	An area of memory in the PGA that contains data such as bind information and run-time memory structures
Privilege	The right to execute a particular type of SQL statement. There are two basic forms of privileges: object and system.
Proactive Tablespace Monitoring	A feature of Oracle Database 10g that manages tablespace disk space usage
Program Global Area	Private memory area for use by a process
Recovery catalog	A separate database that keeps historical data concerning backup activities
Recovery Manager	The Oracle utility used to back up and restore database files
Recycle bin	A data dictionary table that maintains the relationships between the original names of dropped objects and their system-generated names
Recycle buffer cache	A region of memory in the SGA, which holds data that is quickly aged out of the buffer cache

Term	Definition
Redo log buffer	A region of memory that caches redo information until it can be written to disk
Redo Log File Sizing Advisor	A feature of Enterprise Manager that offers redo log file-sizing advice
Resource Manager	A feature of the Oracle database that gives the Oracle database server more control over resource management decisions, thus circumventing problems resulting from inefficient operating system management
Resumable space allocation	A means for suspending, and later resuming, the execution of large database operations in the event of space allocation failures
RMAN	Recovery Manager
RMAN Repository	A storage structure that maintains metadata about a database's backup and recovery operations
Scheduler	A new database feature that enables database administrators and application developers to control when and where various tasks take place in the database environment
SCN	System change number
Segment Advisor	An advisor that monitors object space issues and analyzes growth trends
Segment Resource Estimator	The new segment resource estimation feature that enables you to estimate the amount of resources that the creation of a new segment would require
Server sessions	The server processes (UNIX) or threads (Windows NT/2000) invoked by a client utility to connect to the target database
Session memory	Memory in the PGA that is allocated to hold session variables and other information related to the session
SGA	System Global Area. It is the memory area shared by all server and background processes.
SGA Advisor	An advisor that makes recommendations for SGA-related parameter settings
Shared pool	A region of memory that caches various constructs that can be shared among users
Shrink Advisor	See the Segment Advisor.
SID	System Identifier. It defaults to the database name and uniquely identifies the instance on a given server.
SQL	Structured Query Language
SQL Access Advisor	A tool that determines optimal data access path (for example, the use of indexes and materialized views)
SQL Tuning Advisor	A tool that provides tuning advice for SQL statements
Statspack	A set of SQL, PL/SQL, and SQL*Plus scripts that allow the collection, automation, storage, and viewing of performance data. This feature has been replaced by the Automatic Workload Repository.
Streams pool	An optional region of memory in the SGA that is used by Oracle

Term	Definition
Streams	Streams
System statistics	Statistics that describe the system's hardware characteristics, such as I/O and CPU performance and utilization, to the query optimizer
Tablespace	A logical grouping of data files
Target database	The database that you are attempting to connect to
Threshold	A boundary value against which metric values are compared
Undo Advisor	A feature of Enterprise Manager that suggests parameter values and the amount of additional space that is needed to support flashback for a specified time
Undo data	A copy of original data stored whenever a DML transaction changes data. Undo data is used to roll back a transaction and to provide read-consistent views of changing data.
User Global Area	An area of memory within the shared pool or large pool that contains the session information for the Oracle shared server sessions
UTC	Universal Time Coordinates. This is a global time stamp in the Uniform Audit Trail.
VPD	Virtual Private Database
Wait event	Statistics that are incremented by a server process or thread to indicate that the process had to wait for an event to complete before being able to continue processing
Work area	A private allocation of memory in the PGA, used for sorts, hash joins, and other operations that are memory intensive
Workload repository	See AWR.

Oracle Restart

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Objectives

After completing this lesson, you should be able to use Oracle Restart to manage components.



Copyright © 2009, Oracle. All rights reserved.

Oracle Restart

Oracle Restart implements a high availability solution for stand-alone Oracle databases.

- Can monitor and restart the following components:
 - Database instances
 - Oracle Net listener
 - Database services
 - Automatic Storage Management (ASM) instance
 - ASM disk groups
 - Oracle Notification Services (ONS/eONS)
- Runs periodic check operations to monitor the health of the components
- Runs out of the Oracle Grid Infrastructure home, which you install separately from Oracle Database homes

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Restart

Oracle Restart is designed to improve the availability of your Oracle Database. It implements a high availability solution for single instance (nonclustered) environments only. For Oracle Real Application Cluster (Oracle RAC) environments, the functionality to automatically restart components is provided by Oracle Clusterware. Oracle Restart can monitor the health and automatically restart the following components:

- Database instances
- Oracle Net listener
- Database services
- ASM instance
- ASM disk groups
- Oracle Notification Services (ONS/eONS): Service for sending Fast Application Notification (FAN) events to integrated clients upon failover. The eONS is used by Oracle Enterprise Manager to receive notification of change in status of components managed by Oracle Restart.

Restarting an ASM disk group means mounting it. The ability to restart ONS is applicable only in Oracle Data Guard installations for automatic failover of connections between primary and standby databases through FAN.

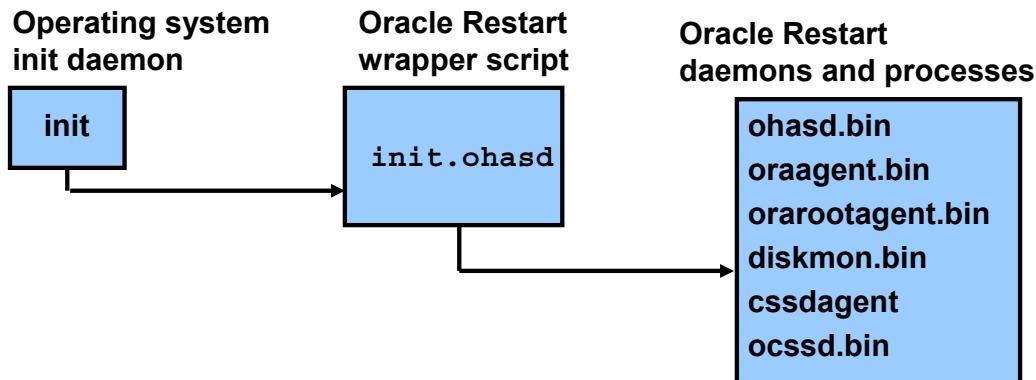
Oracle Restart (continued)

Oracle Restart ensures that the components are started in the proper order, in accordance with component dependencies. If a component must be shut down, it ensures that the dependent components are cleanly shut down first.

Oracle Restart runs out of the Oracle Grid Infrastructure home, which you install separately from Oracle Database homes.

Oracle Restart Process Startup

- Oracle Restart is started by the OS init daemon.



- The Oracle Restart installation modifies the /etc/inittab file to ensure start up every time the machine starts.

```
# cat /etc/inittab
...
h1:35:respawn:/etc/init.d/init.ohasd run >/dev/null 2>&1 </dev/null
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Restart Process Startup

During the installation of Oracle Restart, entries to start a wrapper script are placed in the /etc/inittab operating system file. The wrapper script sets up the environment variables and then starts the Oracle Restart daemons and processes.

When a command is used to stop Oracle Restart, the daemons will be stopped, but the wrapper script process will remain running.

The format of the UNIX /etc/inittab file is as follows:

```
id : run levels : action : process with parameters
```

The wrapper script is started with the respawn action so it will be restarted whenever it is terminated. In addition, the respawn action causes the init process to restart the daemons if they fail.

Some of the Oracle Restart daemons will be running under the root user with real-time priority, and others will be running under the Grid Infrastructure owner with user-mode priorities after they are started. On a Windows platform, operating system services are used instead of wrapper initialization scripts and the daemons are executable binaries.

Note: Executing the wrapper script directly is not supported.

Controlling Oracle Restart

The CRSCTL utility can be used to control the state of Oracle Restart.

- To display the Oracle Restart configuration:

```
$ crsctl config has
```

- To enable or disable the automatic restart of Oracle Restart:

```
$ crsctl [ enable | disable ] has
```

- To start or stop Oracle Restart:

```
$ crsctl [ start | stop ] has
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Controlling Oracle Restart

The CRSCTL utility can be used to control the state of Oracle Restart. It can be used to determine if the autostart capability is enabled or disabled as follows:

```
$ crsctl config has
CRS-4622: Oracle High Availability Services autostart is
enabled.
```

During the installation of Oracle Grid Infrastructure, entries are placed in the /etc/inittab operating system file to start a wrapper script as follows:

```
h1:35:respawn:/etc/init.d/ohasd run >/dev/null 2>&1
</dev/null
```

The wrapper script is responsible for setting up environment variables and then starting the Oracle High Availability Services Daemon (ohasd) along with other related processes. The wrapper script is executed as the `root` user.

When the CRSCTL utility is used to disable the automatic restart of the Oracle Grid Infrastructure, the entry in the /etc/inittab file is not removed. The control files for ohasd are used to control the state of ohasd and determine if automatic restart is enabled or disabled. These are known as `SCLS_SCR` files. For Linux, the location of the control files is defined to be:

```
/etc/oracle/scls_scr/$HOST/<Oracle Restart owner> and
/etc/oracle/scls_scr/$HOST/root
```

Controlling Oracle Restart (continued)

If the CRSCTL utility is used to stop Oracle Restart, all components currently managed by Oracle Restart will also be stopped.

```
$ crsctl stop has
CRS-4549: Stopping resources.
CRS-2673: Attempting to stop 'ora.diskmon' on 'host01'
CRS-2673: Attempting to stop 'ora.DATA.dg' on 'host01'
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host01'
CRS-2677: Stop of 'ora.DATA.dg' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host01'
CRS-2675: Stop of 'ora.diskmon' on 'host01' succeeded
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host01' succeeded
CRS-2677: Stop of 'ora.asm' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host01'
CRS-2677: Stop of 'ora.cssd' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.diskmon' on 'host01'
CRS-2677: Stop of 'ora.diskmon' on 'host01' succeeded
CRS-4133: Oracle High Availability Services has been stopped.
```

In the example above, there is no Oracle Database software installed. If Oracle databases had been registered with Oracle Restart or they had been using ASM, they would have been stopped also.

When starting Oracle Restart with the CRSCTL utility, each component that is started is not displayed to standard output.

```
$ crsctl start has
CRS-4123: Oracle High Availability Services has been started.
```

Note: Invoking the wrapper script directly to start the Oracle Grid Infrastructure processes is not supported.

Choosing the Correct SRVCTL Utility

- Invoke the Server Control (SRVCTL) utility from the Oracle Grid Infrastructure home when working with ASM instances, disk groups, listeners, and ONS.

```
$ export ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid
$ $ORACLE_HOME/bin/srvctl command component options
```

- Invoke the SRVCTL utility from the Oracle Database home when working with the database or database instance.

```
$ export
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1
$ $ORACLE_HOME/bin/srvctl command component options
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Choosing the Correct SRVCTL Utility

Oracle Restart includes the SRVCTL utility that you use to start, stop, and manage Oracle Restart components. After the Oracle Database software is installed in addition to the Oracle Grid Infrastructure software, there will be a copy of the SRVCTL utility in each ORACLE_HOME location. You need to determine the correct ORACLE_HOME location in which to run the SRVCTL utility. You need to run the SRVCTL utility from the Grid Infrastructure software home directory when managing the ASM instance, ASM disk groups, Oracle Net listeners, and the ONS. You need to run the SRVCTL utility from the Oracle Database software home directory when managing the Oracle database instances. To determine the currently mapped location of the SRVCTL utility, use the which command as follows:

```
$ which srvctl
/u01/app/oracle/product/11.2.0/grid/bin/srvctl
```

Note: For the Oracle Net listener, the assumption is that Oracle Grid Infrastructure was installed before the Oracle Database software. If Oracle Restart is added to an existing Oracle Database installation, the Oracle Net listener could be running from the Oracle Database home directory. In that case, you should use the SRVCTL utility from the Oracle Database home to manage the Oracle Net listener.

Oracle Restart Configuration

Oracle utilities will automatically update the Oracle Restart configuration.

Create operations and the Oracle Restart configuration	Automatically added to configuration?
Create a database with OUI or DBCA	YES
Create a database with SQL statement	NO
Create an ASM instance with OUI, DBCA, or ASMCA	YES
Create a disk group (any method)	YES
Add a listener with NETCA	YES
Create a database service with SRVCTL	YES
Create a database service by modifying SERVICE_NAMES initialization parameter	NO
Create a database service with DBMS_SERVICE.CREATE_SERVICE	NO

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Restart Configuration

Oracle Restart maintains a list of all the components that it manages, and maintains configuration information for each component. All of this information is collectively known as the “Oracle Restart configuration.” When Oracle Restart is installed, many operations that create Oracle components using Oracle utilities will automatically add the components to the Oracle Restart configuration. If a component is created manually without using an Oracle utility, then SRVCTL commands can be used to add it to the Oracle Restart configuration if desired. The table in the above slide shows which create operations automatically add the component to the Oracle Restart configuration and which create operations do not update the Oracle Restart configuration.

Note: The same principal applies to delete operations.

Using the SRVCTL Utility

- The SRVCTL utility is used to start, stop, and manage Oracle Restart components with the following syntax:

```
$ srvctl command component options
```

- The following command and components are supported:

Commands	add config disable enable getenv modify remove setenv start status stop unsetenv
Components	asm db dg filesystem home lsnr serv ons eons

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Using the SRVCTL Utility

When Oracle Restart is in use, Oracle strongly recommends that you use the SRVCTL utility to start, stop, and manage all Oracle Restart components. The SRVCTL utility is recommended for the following reasons:

- All dependencies between components are maintained. This enables Oracle Restart to start or to stop any dependent components first.
- Components are started according to their Oracle Restart configuration.
- Environment variables stored in the Oracle Restart configuration for the components are set.

Oracle Restart components can also be started with other utilities such as the listener control (LSNRCTL) utility or SQL*Plus, but the benefits listed above may not be obtained with other utilities. The SRVCTL utility with Oracle Restart supports 12 commands and 9 components. The options that are allowed vary with each command and component combination. The SRVCTL utility syntax is as follows:

```
srvctl command component options
```

where:

- *command* is a verb such as `start`, `stop`, or `remove`
- *component* is the object on which SRVCTL performs the command, such as a database
- *options* extend the use of the preceding command to include additional parameters

Obtaining Help for the SRVCTL Utility

The SRVCTL utility provides online help for its commands, components, and options.

- For help with general usage:

```
$ srvctl -h
```

- For help on a particular command:

```
$ srvctl command -h
```

- For help on a particular command and component:

```
$ srvctl command component -h
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Obtaining Help for the SRVCTL Utility

The SRVCTL utility provides detailed online help for its commands, components, and options. To display the online Help, you use the help (-h) option to display usage information. If the help (-h) option is the only parameter specified, SRVCTL displays a general outline of all commands with the most common options used for each command and component combination. This will not be a complete list of all supported options. For more detailed and complete information, the help (-h) option can be used for a specific command, or for a specific command and component combination.

Starting Components by Using the SRVCTL Utility

Oracle recommends that the SRVCTL utility be used to start all components.

- Examples of starting individual components:

```
$ srvctl start database -d PROD -o mount
$ srvctl start listener -l crmlistener
$ srvctl start service -d PROD -s "service1,service2"
$ srvctl start diskgroup -g "DATA,FRA"
$ srvctl start asm
$ srvctl start eons -v
$ srvctl start ons
```

- Example of starting all Oracle Restart components in a specified Oracle home:

```
$ srvctl start home -o oracle_home -s state_file
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Starting Components by Using the SRVCTL Utility

The SRVCTL utility can be used to start individual components, along with any dependent components that are necessary. For example, the `srvctl start database -d PROD` command may also start the listener, the ASM instance, and multiple disk groups if those components have been defined as being managed by Oracle Restart and are listed as dependent components to the PROD database.

The SRVCTL utility can also be used to start all components that are associated with a specified Oracle home and have been configured for Oracle Restart with the following command:

```
srvctl start home -o /u01/app/oracle/product/11.2.0/dbhome_1
-s /usr/local/bin/group_state_file
```

The state file contains the current state information for the components in the Oracle home and is created when the `srvctl status home` command is executed. It is indicated with the state file option (-s), and must specify the complete path of the state file. The state file can be created in any directory.

Note: The options shown in the slide represent the most common options and are not a complete list. You can use the help option (-h) for a complete list of all available options for each command.

Stopping Components by Using the SRVCTL Utility

Oracle recommends that the SRVCTL utility be used to stop all components.

- Examples of stopping individual components:

```
$ srvctl stop database -d PROD -o transactional
$ srvctl stop listener -l crmlistener -f
$ srvctl stop service -d PROD -s "service1,service2"
$ srvctl stop diskgroup -g "DATA,FRA" -f
$ srvctl stop asm -o immediate -f
$ srvctl stop eons -v
$ srvctl stop ons
```

- Example of stopping all Oracle Restart components in a specified Oracle home:

```
$ srvctl stop home -o oracle_home -s state_file -f
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Stopping Components by Using the SRVCTL Utility

The SRVCTL utility can be used to stop individual components, along with any dependent components that must be stopped. For example, the `srvctl stop diskgroup -g "DATA -f"` command will force an unmount of the diskgroup even if files are open in it. It will also stop all database instances that depend on the DATA disk group.

The SRVCTL utility can also be used to stop all components that are associated with a specified Oracle home and have been configured for Oracle Restart with the following command:

```
srvctl stop home -o /u01/app/oracle/product/11.2.0/dbhome_1 -
s /usr/local/bin/group_state_file -f
```

This can be very useful when it is necessary to stop all components such as when you need to apply a patch to the software binaries.

Note: The options shown in the slide represent the most common options and are not a complete list. You can use the help option (-h) for a complete list of all available options for each command.

Viewing Component Status

- You can use the `status` command to view the running status for any component managed by Oracle Restart.
- Display the running status for a database:

```
$ srvctl status database -d orcl
Database is running.
```

- Display the listener status:

```
$ srvctl status lsnr
Listener LISTENER is enabled
Listener LISTENER is running on node(s): host01
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Viewing Component Status

You can use the SRVCTL utility to view the running status (running or not running) for any component managed by Oracle Restart. Additional information is displayed for some components.

The format of the command is as follows:

```
srvctl status object [options]
```

object can be one of the following:

- `asm`: ASM instance
- `db`: Database instance
- `dg`: Oracle ASM disk group
- `filesystem`: Oracle ASM file system
- `home`: Oracle home or Oracle Clusterware home
- `lsnr`: Oracle Net listener
- `ons, eons`: Oracle Notification Services
- `serv`: Database service

Refer to the *Oracle Database Administrators Guide* for a list of options for each of the above objects.

Displaying the Oracle Restart Configuration for a Component

- You can use the `config` command to display the Oracle Restart configuration of a component.
- Display the Oracle Restart configuration for a database:

```
$ srvctl config database -d orcl
Database unique name: orcl
Database name: orcl
Oracle home: /u01/app/oracle/product/11.2.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/orcl/spfileorcl.ora
Domain: example.com
Start options: open
Stop options: immediate
Database role:
Management policy: automatic
Disk Groups: DATA,FRA
Services: east,sales
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Displaying the Oracle Restart Configuration for a Component

You can use the SRVCTL utility to display the Oracle Restart configuration of a component with the `config` command. The `config` command is valid for the `database`, `service`, `asm`, `listener`, `ons`, and `eons` components.

The configuration for an Oracle Restart component can be modified with the SRVCTL utility `modify` command. The following syntax shows an example of how to modify the database with the unique name of `orcl` to use a different, nonstandard directory for the server parameter file (SPFILE).

```
srvctl modify database -d orcl -p
/usr/local/oracle/spfileorcl.ora
```

Manually Adding Components to the Oracle Restart Configuration

Components can be manually added to the Oracle Restart configuration with the `add` command.

- To define a new listener that was not created with NETCA, use the following syntax:

```
$ srvctl add listener -l MYLISTENER -p TCP:1525 -o
/u01/app/oracle/product/11.2.0/grid
```

- To specify a nondefault location for the new listener's networking files, use the following syntax:

```
$ srvctl setenv listener -l MYLISTENER -t
TNS_ADMIN=/usr/local/oracle
```

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Manually Adding Components to the Oracle Restart Configuration

The SRVCTL utility can be used to manually add components to the Oracle Restart configuration with the `add` command. If the component was created with an Oracle utility such as NETCA, DBCA, ASMCA, or OUI, it would have been automatically added to the Oracle Restart configuration and it would not be necessary to manually add it.

The slide illustrates manually adding a new listener called `MYLISTENER` to the Oracle Restart configuration. The listener will use the software binaries of the Grid home installation, but will depend on a nonstandard location for the networking files. For this example, it is assumed that the `listener.ora` networking file has been created in the `/usr/local/oracle` directory. The `setenv` command of the `srvctl` utility is used to define environment variables that may be needed for specific components. The `TNS_ADMIN` environment variable is set to the nondefault location of the `listener.ora` file, and is defined only for the listener named `MYLISTENER`. This will not have an impact on any other listeners that may already exist and that use different directories for the networking files.

The SRVCTL utility includes a `remove` command to manually delete a component from the Oracle Restart configuration. To delete the listener created above, use the following syntax:

```
srvctl remove listener -l mylistener -f
```

This will also remove the environment variable that was associated with the listener.

Quiz

You invoke the SRVCTL utility from the Oracle Grid Infrastructure home when working with:

1. Listeners
2. ASM instances
3. Database instances
4. ASM disk groups

 ORACLE

Copyright © 2009, Oracle. All rights reserved.

Answers: 1, 2, and 4

Summary

In this lesson, you should have learned how to use Oracle Restart to manage components.



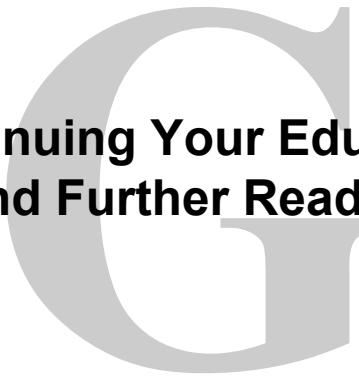
Copyright © 2009, Oracle. All rights reserved.

Practice 3-1: Overview

In this practice, you use Oracle Restart to manage components.



Copyright © 2009, Oracle. All rights reserved.



Continuing Your Education and Further Reading

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Where Do You Go from Here?

“To stay competitive in the tech industry, never stop learning. Always be on the lookout for better ways of doing things and new technologies. Our industry does not reward people who let themselves stagnate.”

—John Hall, Senior Vice President, Oracle University

This appendix provides a few resources to help you with continuing your education.

ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Continuing Education Resources

Resources to learn more about working with Oracle Database 11g include:

- Oracle University Web site
- Oracle University Knowledge Center
- Oracle Technology Network:
 - Oracle by Example
 - *Oracle Magazine*
 - Oracle Database product page
- Technical support: My Oracle Support
- Oracle Database product page



Copyright © 2009, Oracle. All rights reserved.

Oracle University



The screenshot shows the Oracle University homepage. At the top is the Oracle University logo with 'ORACLE' in blue and 'UNIVERSITY' in red. Below it is a 'UNITED STATES' link. The main content area has a red header 'ORACLE UNIVERSITY'. Underneath are two columns of links: '100% Student Satisfaction', 'Course Schedule', 'Knowledge Center', 'Self-Study CD-ROM', and 'User Adoption Services' in the first column; and 'PRODUCT COURSES' in the second column, followed by a list of course categories including 'Database and Grids', 'Fusion Middleware', 'Development Tools', 'Collaboration', 'Data Warehouse', 'Linux | Java', 'E-Business Suite', 'PeopleSoft Enterprise', 'JD Edwards EnterpriseOne', 'JD Edwards World', 'Siebel', 'Retail Industry', 'Telecom Industry', and 'Utilities Industry'.

Learn Oracle from Oracle! No one knows Oracle technology better than Oracle University.

- Worldwide education services
- 100% student satisfaction
- Learn with the format that best suits *your* needs:
 - Instructor-Led Inclass Training
 - Live Web Class
 - Self-Study CD-ROMs
- Certification

Oracle University

<http://www.oracle.com/education>

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle University

Oracle University is the world's largest corporate educator with education centers around the globe. The goal is 100% student satisfaction.

Oracle certifications are tangible, industry-recognized credentials that provide measurable benefits to IT Professionals and their employers. Numerous certification paths exist, for example, for DBAs:

- Oracle Certified Associate (OCA)
- Oracle Certified Professional (OCP)
- Oracle Certified Master (OCM), and
- Specialty certifications, for example, Oracle 10g: Managing Oracle on Linux Certified Expert

Continuing Your Education

- Database specialty classes:
 - *Oracle Database 10g: Administration Workshop II*
 - *Oracle Database 11g: Performance Tuning*
 - *Oracle Database 11g: Security*
- Other specialty classes:
 - *Oracle 11g: RAC and Grid Foundation Administration*
 - *Oracle Database 11g: RAC Administration*

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Continuing Your Education

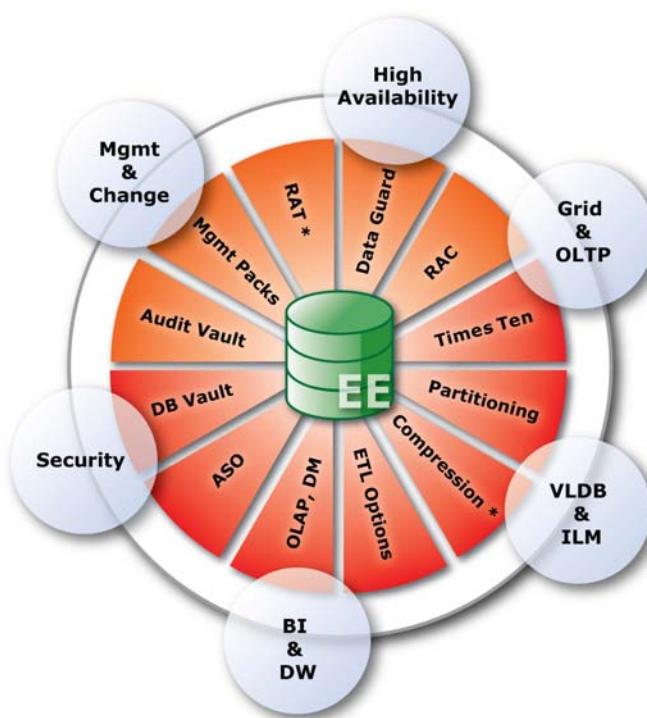
The *Oracle Database 10g: Administration Workshop II* course provides additional training as a database administrator. You cover advanced database recovery strategies, performance monitoring and tuning, and distributed data concepts. Oracle recommends that you complete this course before beginning specialty courses. Additional courses are available on topics that can enhance your abilities as an Oracle database administrator, such as:

- *Oracle Database 11g: Performance Tuning*
- *Oracle Database 11g: Security*

Other specialty courses include *Oracle 11g: RAC and Grid Foundation Administration* and *Oracle Database 11g: RAC Administration*.

Consult the Oracle University Web site for an up-to-date list of all courses. Consult the Oracle University Learning Paths for more suggested courses for your job description.

Database Specialty Areas



Modern Enterprise Grids

- Real Application Clusters
- Management packs
- TimesTen In-Memory Database

Information Lifecycle Management

- Partitioning
- Advanced Compression

Data Warehousing

- Oracle Information Appliances
- OLAP, Mining, Warehouse Builder

Governance, Risk & Compliance

- Security Options
- Total Recall

Change management

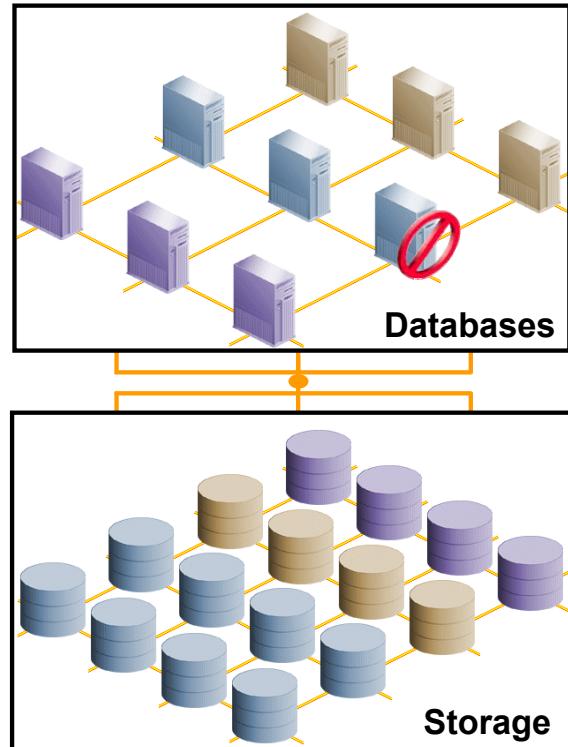
- Real Application Testing

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Real Application Clusters

- Consolidating different workloads to a single grid
- Virtualizing the information platform
- Flexible physical infrastructure (including dedicated servers)



ORACLE

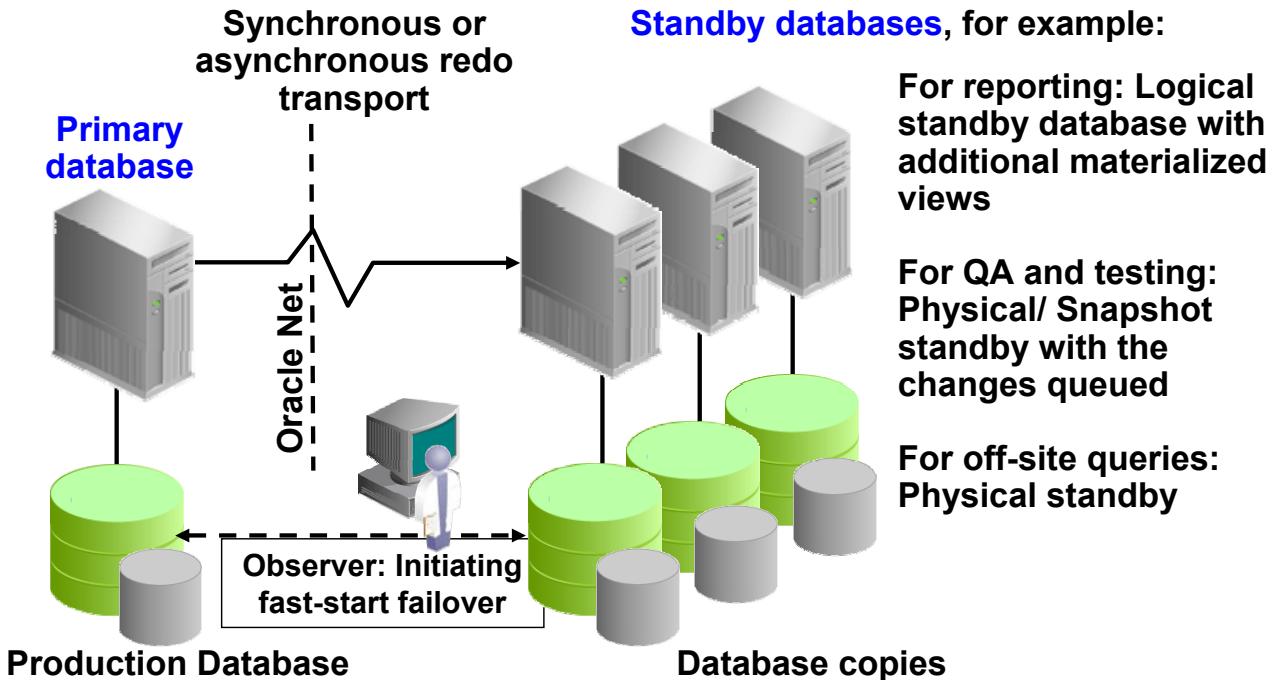
Copyright © 2009, Oracle. All rights reserved.

Why use RAC?

Oracle Real Application Clusters (RAC) enables high utilization of a cluster of standard, low-cost modular servers such as blades. RAC offers automatic workload management for services. Services are groups or classifications of applications that comprise business components corresponding to application workloads. Services in RAC enable continuous, uninterrupted database operations and provide support for multiple services on multiple instances. You assign services to run on one or more instances, and alternate instances can serve as backup instances. If a primary instance fails, Oracle moves the services from the failed instance to a surviving alternate instance. Oracle also automatically load-balances connections across instances hosting a service.

RAC harnesses the power of multiple low-cost computers to serve as a single large computer for database processing, and provides the only viable alternative to large-scale SMP boxes for all types of applications. RAC, which is based on a shared-disk architecture, can grow and shrink on demand without the need to artificially partition data among the servers of your cluster. RAC also offers a single-button addition and removal of servers to a cluster. Thus, you can easily provide or remove a server to or from the database.

Oracle Data Guard



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Data Guard

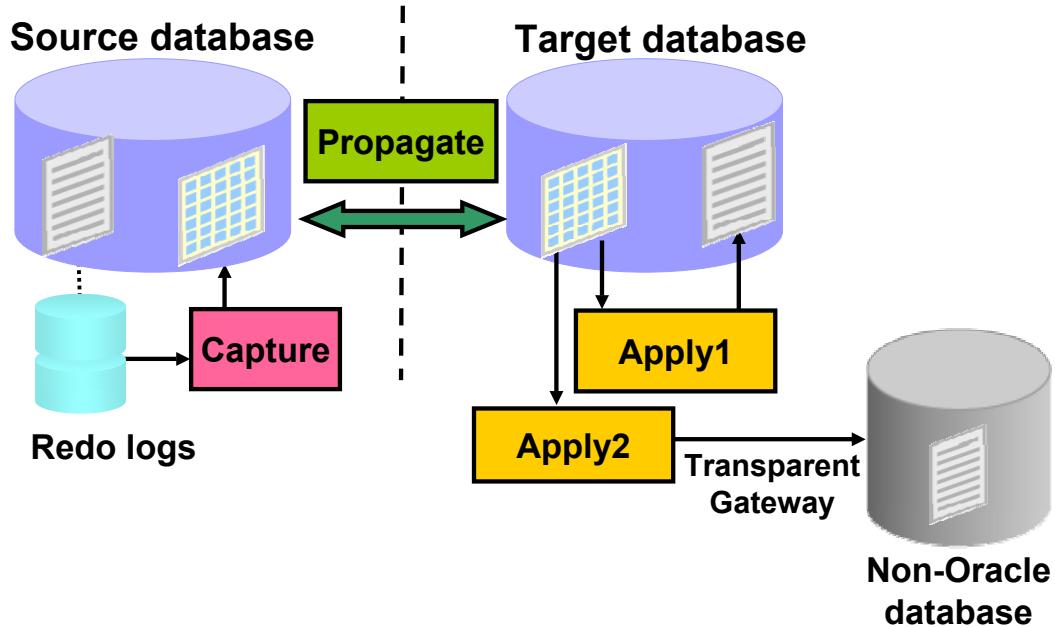
Oracle Data Guard is a management, monitoring, and automation software infrastructure that works with a production database and one or more standby databases to protect your data against failures, errors, and corruptions that might otherwise destroy your database. It protects critical data by providing facilities to automate the creation, management, and monitoring of the databases and other components in a Data Guard configuration. It automates the process of maintaining a copy of an Oracle production database (called a *standby database*) that can be used if the production database is taken offline for routine maintenance or becomes damaged.

In a Data Guard configuration, a production database is referred to as a *primary database*. A *standby database* is a synchronized copy of the primary database. Using a backup copy of the primary database, you can create from one to nine standby databases. The standby databases, together with the primary database, make up a Data Guard configuration. Each standby database is associated with only one primary database.

Note: You can use the Cascaded Redo Log Destinations feature to incorporate more than nine standby databases in your configuration.

Configuring standby redo log files is highly recommended on all standby databases in a Data Guard configuration, including the primary database to aid in role reversal.

Streams Overview



ORACLE

Copyright © 2009, Oracle. All rights reserved.

Streams Overview

A stream is a flow of information either within a database or from one database to another. Oracle Streams is a set of processes and database structures that enable you to share data and messages in a data stream. The unit of information that is put into a stream is called an event:

- DDL or DML changes, formatted as an LCR
- User-created events

Events are staged in and propagated between queues.

Most people think of Streams as replication where all databases can be updatable, and without platform or release considerations. Characteristics include:

- All sites: Active and updateable
- Automatic conflict detection and optional resolution
- Supporting data transformations
- Flexible configurations: n-way, hub & spoke, and so on
- Different database platforms, releases and schemas
- Providing high availability for applications (where update conflicts can be avoided or managed)

Oracle Streams: Basic Elements

By using Oracle Streams, you can share data and events in a data stream, either within a database or from one database to another.

Oracle Streams uses queues to stage events for propagation or consumption. You can use Oracle Streams to propagate events from one queue to another, and these queues can be in the same database or in different databases. You may stage two types of events in a queue used by Streams: captured events (logical change records, or LCRs) and user-enqueued events (which can be messages or LCRs):

- Changes to the database can be captured from the redo logs. You can then format these changes into LCRs. The LCRs can represent data manipulation language (DML) or data definition language (DDL) changes. The database where changes are generated in the redo log is called the source database.
- You can also enqueue user events explicitly with a user application. These explicitly enqueued events can be LCRs or user-created messages. A message is the smallest unit of information that is inserted into and retrieved from a queue. A message consists of data as well as information to govern the interpretation and use of the message data.

You can divide Oracle Streams into a small set of tasks. By configuring these tasks, you can control what information is put into a stream, how the stream flows from node to node, what happens to events in the stream as they flow into each node, and how the stream terminates.

You can customize each task to address specific requirements and business needs. The result is a new feature that provides greater functionality and flexibility than traditional solutions for capturing and managing events, and for sharing the events with other databases and applications. Oracle Streams provides the capabilities that are needed to build and operate distributed enterprises and applications, data warehouses, and high-availability solutions.

The three basic tasks of Oracle Streams are:

- **Capture:** To capture DML or DDL events automatically from the redo log. User-created events are not captured automatically but are placed into a queue via an explicit enqueue operation.
- **Staging:** To store and propagate events between databases. Propagation can be performed explicitly if needed.
- **Apply:** To apply DML or DDL events to a destination database or to pass the events to an application.

You can perform these tasks in a single database or combine them with tasks in other databases to form a distributed environment.

Multi-Database Streams

Events propagate between the staging areas in each database. The capture and consumption elements can be active in any database. For example, you can configure bidirectional data replication with a capture process, propagation job, and apply process at each site. Or, you can have a single-source system with capture and propagation at one site and apply at several other databases. You can also have an arbitrary number of databases. Some of the more complex environments may need hundreds of databases sharing information with Oracle Streams.

Oracle Technology Network

Oracle Technology Network is a *free* resource with information about the core Oracle software products, including database, Application Server, Collaboration Suite, and development tools. You can have access to:

- Technology centers
- Discussion forums
- Software downloads
- Online documentation
- Oracle by Example
- Code samples

... and much more!



<http://www.oracle.com/technology>

ORACLE

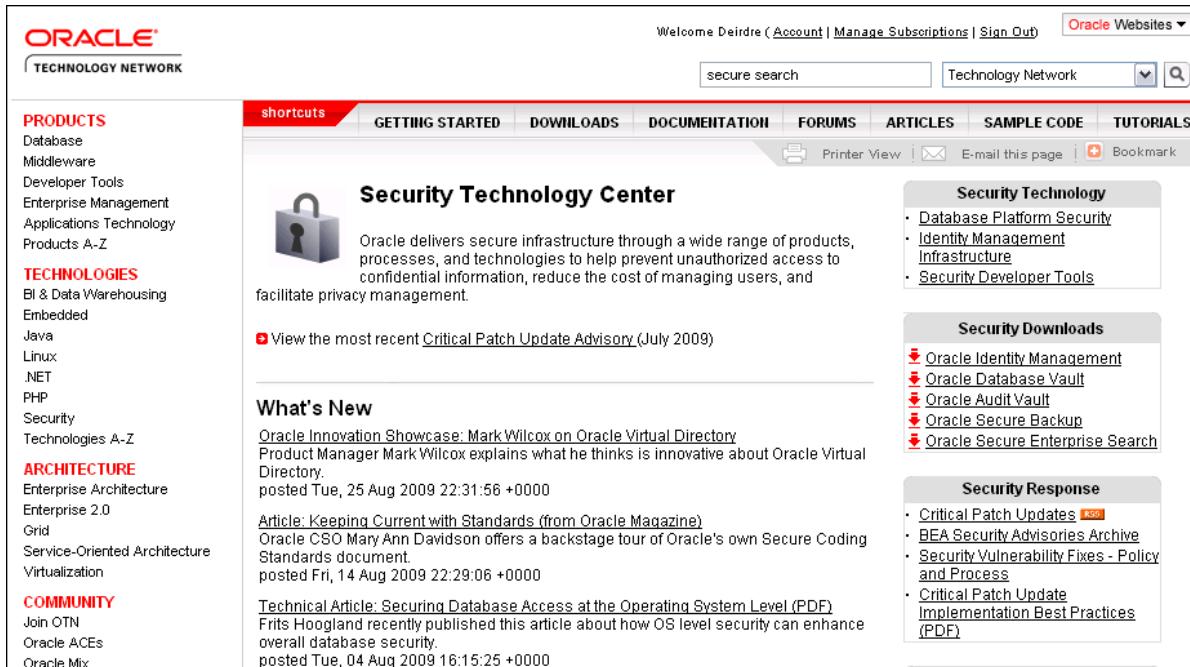
Copyright © 2009, Oracle. All rights reserved.

Oracle Technology Network

Oracle Technology Network (OTN) hosts the latest news about Oracle technology and products. Additionally, OTN provides peer-to-peer forums, white papers, security bulletins, and other vital information for the Oracle professional.

In addition to tips, tricks, and techniques for getting the most out of your Oracle software, you can download that software from OTN. Remember that all software downloads are free; each comes with a development license that allows you to use full versions of the products only when developing and making prototypes of your applications.

Security



The screenshot shows the Oracle Technology Network homepage with a red banner at the top. The main content area is titled 'Security Technology Center' and features a lock icon. It discusses Oracle's secure infrastructure and links to the most recent Critical Patch Update Advisory (July 2009). Below this, there are sections for 'What's New' (including an Oracle Innovation Showcase and an Article on Secure Coding Standards), and 'Community' (Join OTN, Oracle ACEs, Oracle Mix). On the right, there are three boxes: 'Security Technology' (Database Platform Security, Identity Management, Infrastructure, Security Developer Tools), 'Security Downloads' (links to Oracle Identity Management, Database Vault, Audit Vault, Secure Backup, and Secure Enterprise Search), and 'Security Response' (links to Critical Patch Updates, BEA Security Advisories Archive, Security Vulnerability Fixes, and Critical Patch Update Implementation Best Practices). The top right of the page shows a welcome message for Deirdre and links to Account, Manage Subscriptions, and Sign Out, along with an Oracle Websites dropdown.

<http://www.oracle.com/technology/deploy/security/index.html>

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Security

For more information about all security-related aspects of the database, visit the “Security Technology Center,” which is updated regularly.

Oracle by Example

- What is an OBE?
 - A set of hands-on, step-by-step instructions
- Where can I find them?
 - <http://www.oracle.com/technology/obe>
- What is available?
 - Hundreds of OBE tutorials on many of the Oracle product areas



Copyright © 2009, Oracle. All rights reserved.

Oracle by Example

The Oracle by Example (OBE) series provides hands-on, step-by-step instructions on how to use various new features of Oracle products. OBEs help to reduce the time spent on learning a new product capability and enhance the users' understanding of how the feature can be implemented in their environment. Currently, OBEs are available for the Oracle database, Fusion Middleware, Oracle Application Server, Oracle Enterprise Manager Grid Control, Oracle Collaboration Suite, JDeveloper, and Business Intelligence. OBEs can be accessed at <http://www.oracle.com/technology/obe>.

Oracle Magazine

- Free subscription
- Oracle Magazine Archives
<http://www.oracle.com/technology/oramag/index.html>



ORACLE®

Copyright © 2009, Oracle. All rights reserved.

Oracle Magazine

Among the many different types of resources to which you have access from OTN is *Oracle Magazine*. You can receive your free subscription also by mail.

Oracle Applications Community

Oracle Technology Network is a resource for Oracle Applications users and implementers. You can have access to:

- Discussion forums
- User groups
- Online chat
- Documentation
- Training
- Upgrade information
 - ... and much more!



<http://www.oracle.com/technology/community/apps/index.html>

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Technical Support: My Oracle Support

Access to My Oracle Support is included as part of your annual support maintenance fees. In addition to the most up-to-date technical information available, My Oracle Support gives you access to:

- Service requests (SRs)
- Certification matrices
- Technical forums monitored by Oracle experts
- Software patches
- Bug reports



<http://metalink.oracle.com>



Copyright © 2009, Oracle. All rights reserved.

Oracle MetaLink

My Oracle Support is your gateway to Oracle's Support resources. Here you find answers to the most common issues facing Oracle administrators and developers, as well as resources to solve many of those issues.

Like Oracle Technology Network, My Oracle Support includes the most recent headlines about issues that affect the Oracle professional.

Oracle Database Product Page

From the Oracle Database product page on OTN, there are links to:

- Database Focus Areas
- White Papers
- Option Data Sheets
- Related technologies
- Discussions
- Other useful resources

ORACLE

Copyright © 2009, Oracle. All rights reserved.

Oracle Database Product Page

From the Oracle Database product page on OTN, there are links to useful information such as:

- **Database Focus Areas:** Provides links to other database-related focus areas such as Grid/Clustering, High Availability, Performance, and Manageability
- **White Papers:** Provides detailed information to help you learn more about the Oracle Database 11g product family
- **Option Data Sheets:** Provides detailed information about the various Oracle Database 11g options
- **Related technologies:** Provides links to related technologies such as Oracle Application Express and Oracle SQL Developer
- **Discussions:** Provides links to ongoing discussions related to Oracle Database 11g
- **Other resources:** Links to documentation, forums, links for learning more, and other useful resources

The documents here are live documents that are updated whenever the need arises, as well as new documents that may be added at any time. Check back regularly to find the latest information available.

Thank You!

We hope your experience with Oracle University has been enjoyable. We welcome your feedback on how we can improve to better meet your needs:

- End-of-course evaluations
- Oracle University Office of Customer Satisfaction
- Oracle Education Services

We hope to see you in class again soon.



Copyright © 2009, Oracle. All rights reserved.

Thank You!

Oracle University's mission is to enhance the adoption of Oracle technology. Our goal is to partner with you, providing information that is pertinent, timely, and relevant to your needs.

Please take a minute to complete the end-of-course evaluation and let us know how we can serve you better. In the U.S., feel free to e-mail our office of customer satisfaction at:

`customersat_us@oracle.com`

If you have questions about continuing your Oracle education, need help finding a class, or want to arrange for on-site training at your company, contact Oracle Education Services for assistance. In the U.S., dial 800.529.0165. For contact numbers outside the U.S., visit the following Web site:

<http://www.oracle.com/education/index.html?contact.html>

Thanks again. We hope to see you in another class!