

Oracle Database 11g: OCM Exam Preparation Workshop

Student Guide

D69748GC20

Edition 2.0

June 2013

D82216

ORACLE®

Author

Setsuko Fujitani

**Technical Contributors
and Reviewers**

Sharath Bhujani

Joel Goodman

Setsuko Fujitani

Lakshmi Narapareddi

Editors

Vijayalakshmi Narasimhan

Rashmi Rajagopal

Malavika Jinka

Publishers

Sujatha Nagendra

Joseph Fernandez

Copyright © 2013, Oracle and/or its affiliates. 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 and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Contents

1 Configuring for Recoverability

- Objectives 1-2
- Purpose of Backup and Recovery Functionality 1-3
- Typical Backup and Recovery Tasks 1-4
- Oracle Backup and Recovery Solutions 1-6
- Using Recovery Manager 1-7
- Types of RMAN Commands 1-8
- Job Commands: Example 1-9
- Configuring Your Database for Backup and Recovery Operations 1-10
- ARCHIVELOG Mode 1-11
- Configuring ARCHIVELOG Mode 1-12
- Configuring Archive Log Destinations 1-13
- Guaranteeing Archive Log Success 1-14
- Specifying a Backup Destination 1-16
- Specifying a Retention Policy 1-17
- A Recovery Window Retention Policy: Example 1-19
- Using a Flash Recovery Area 1-20
- Defining a Flash Recovery Area 1-22
- Defining a Flash Recovery Area Using Enterprise Manager 1-23
- Flash Recovery Area Space Management 1-24
- Flash Recovery Area Space Usage 1-26
- Monitoring the Flash Recovery Area 1-28
- Benefits of Using a Flash Recovery Area 1-29
- Summary 1-30

2 Configuring Backup Specifications

- Objectives 2-2
- Using RMAN to Create Backups 2-3
- Backup Destinations 2-4
- Configuring Persistent Settings for RMAN 2-5
- Using Enterprise Manager to Configure RMAN Settings 2-6
- Control File Autobackups 2-7
- Managing Persistent Settings 2-9
- Configuring Devices for Backup 2-10
- Configuring and Allocating Channels for Use in Backups 2-12

Configuring Backup Optimization 2-13
Summary 2-15

3 Using RMAN to Create Backups

Objectives 3-2
Creating Backup Sets 3-3
Creating Image Copies 3-4
Creating a Whole Database Backup 3-6
Saving Backup Space with Unused Block Compression 3-8
RMAN Backup Types 3-9
Fast Incremental Backup 3-11
Enabling Fast Incremental Backup 3-12
Monitoring Block Change Tracking 3-13
Creating Duplexed Backup Sets 3-14
Creating Duplexed Backup Sets Using CONFIGURE BACKUP COPIES 3-15
Creating Duplexed Backup Sets Using BACKUP COPIES 3-16
Creating Backups of Backup Sets 3-17
Backing Up Read-Only Tablespaces 3-18
Archival Backups: Concepts 3-19
Creating Archival Backups with EM 3-21
Creating Archival Backups with RMAN 3-22
Managing Archival Database Backups 3-23
Multisection Backups: Overview 3-24
Creating RMAN Multisection Backups 3-25
Compressing Backups 3-26
Encrypting Backups 3-27
Backing Up Recovery Files 3-29
Using a Media Manager 3-30
Performing Proxy Copies 3-32
Creating an Oracle-Suggested Backup 3-33
Managing Backups: Reporting 3-34
Managing Backups: Dynamic Performance Views 3-36
Using Enterprise Manager to View Backup Reports 3-37
Managing Backups: Cross-Checking and Deleting 3-38
Summary 3-39

4 Using RMAN to Perform Recovery

Objectives 4-2
Using RMAN RESTORE and RECOVER Commands 4-3
Performing Recovery Using Enterprise Manager 4-4

Performing Complete Recovery: Loss of a Noncritical Data File in ARCHIVELOG Mode	4-5
Performing Complete Recovery: Loss of a System-Critical Data File in ARCHIVELOG Mode	4-6
Recovering Image Copies	4-7
Recovering Image Copies: Example	4-8
Performing a Fast Switch to Image Copies	4-10
Using SET NEWNAME for Switching Files	4-11
Performing Restore and Recovery of a Database in NOARCHIVELOG Mode	4-12
Creating Restore Points	4-13
Performing Incomplete Recovery	4-14
Performing Recovery with a Backup Control File	4-16
Restoring the Server Parameter File from the Control File Autobackup	4-17
Restoring the Control File from Autobackup	4-18
Using Incremental Backups to Recover a Database in NOARCHIVELOG Mode	4-20
Restoring and Recovering the Database on a New Host	4-21
Preparing to Restore the Database to a New Host	4-22
Restoring the Database to a New Host	4-23
Performing Disaster Recovery	4-27
Summary	4-29

5 SQL Performance Analyzer

Objectives	5-2
Challenges Faced by DBAs When Performing Changes	5-3
Change Is the Only Constant	5-4
Change Management in Oracle Database 11g	5-5
Life Cycle of Change Management	5-6
SQL Performance Analyzer: Overview	5-8
SQL Performance Analyzer: Use Cases	5-9
Using SQL Performance Analyzer	5-10
Step 1: Capture SQL Workload	5-11
Step 2: Transport to a Test System	5-12
Step 3: Build Before Change Performance Data	5-13
Step 4: Implement Planned Change and Step 5: Build After-Change Performance Data	5-14
Step 6: Compare and Analyze Performance and Step 7: Tune Regressed SQL	5-15
Quiz	5-16
Accessing SQL Performance Analyzer	5-17
Using Enterprise Manager to Access SQL Performance Analyzer	5-18
SQL Performance Analyzer: PL/SQL Example	5-19

Tuning Regressed SQL Statements	5-21
Testing Database Upgrades: Oracle9i Database and Oracle Database 10g Release 1	5-22
Testing Database Upgrades: Oracle Database 10g Release 2 and Later Releases	5-25
SQL Performance Analyzer: Data Dictionary Views	5-28
Summary	5-29

6 SQL Plan Management

Objectives	6-2
SQL Plan Management: Overview	6-3
SQL Plan Baseline: Architecture	6-4
Loading SQL Plan Baselines	6-6
Evolving SQL Plan Baselines	6-7
Viewing Important Baseline SQL Plan Attributes	6-8
Important Baseline SQL Plan Attributes	6-9
SQL Plan Selection	6-10
Quiz	6-12
Possible SQL Plan Manageability Scenarios	6-13
SQL Performance Analyzer and SQL Plan Baseline Scenario	6-14
Loading a SQL Plan Baseline Automatically	6-15
Purging SQL Management Base Policy	6-16
Enterprise Manager and SQL Plan Baselines	6-17
Using the MIGRATE_STORED_OUTLINE Functions	6-18
Summary	6-19

7 Grid Control Architecture

Objectives	7-2
Grid Control Architecture	7-3
Oracle Management Service	7-5
Oracle Management Agent	7-6
Oracle Management Repository	7-7
Grid Control Targets	7-8
Grid Control Console: Home	7-9
Grid Control Console: Targets	7-10
Grid Control Console: Deployments	7-11
Grid Control Console: Alerts	7-12
Grid Control Console: Compliance	7-13
Grid Control Console: Jobs	7-14
Grid Control Console: Reports	7-15
Grid Control Console: My Oracle Support	7-16

Grid Control Console: Setup 7-18
Grid Control Console: Preferences 7-19
Grid Control High Availability 7-20
Quiz 7-21
Summary 7-22

8 Grid Control Installation

Objectives 8-2
Installing Grid Control 8-3
Oracle WebLogic Server Installation 8-4
Oracle Installer: Welcome 8-5
Choose Middleware Home Directory 8-6
Register for Security Updates 8-7
Choose Install Type 8-8
Choose Product Installation Directories 8-9
Installation Summary 8-10
Installation Complete 8-11
Installing Grid Control 8-12
My Oracle Support Details 8-13
Check for Updates 8-14
Select Installation Type 8-15
Check Prerequisites 8-16
Specify Install Locations 8-17
Create WebLogic Server Domain 8-18
Connect to Oracle Database 8-19
Configure Oracle Management Repository 8-20
Secure Oracle Management Service 8-21
Customize Ports 8-22
Review 8-23
Execute Configuration Scripts 8-24
Installation Progress Details 8-25
Finish 8-26
Oracle Management Agent Installation 8-27
OMA Installation: Agent Push 8-29
Default Ports Used for Grid Control Installation 8-34
Configuring Firewalls 8-35
Grid Control Installation Directories 8-37
Grid Control Installation Directories: Instance Home 8-38
Grid Control Installation Directories: Agent 8-40
Grid Control Installation Directories: OMS 8-41

Quiz 8-42
Summary 8-43

9 Setting Up Enterprise Manager Grid Control

Objectives 9-2
Grid Control Administrators 9-3
Privileges 9-4
System Privileges 9-5
Target Privileges 9-7
Roles 9-9
Creating a Super Administrator Account 9-10
Creating an Administrator Account 9-12
Maintaining Administrators 9-15
Creating Roles 9-16
Preferences 9-19
Preferred Credentials 9-20
Setting Preferred Credentials 9-21
Managing Target Subtabs 9-22
Enterprise Manager Command Line Interface 9-23
Setting Up EM CLI 9-24
Quiz 9-26
Summary 9-29

10 Introduction to Oracle Data Guard

Objectives 10-2
What Is Oracle Data Guard? 10-3
Types of Standby Databases 10-4
Types of Data Guard Services 10-6
Role Transitions: Switchover and Failover 10-7
Oracle Data Guard Broker Framework 10-8
Choosing an Interface for Administering a Data Guard Configuration 10-9
Oracle Data Guard: Architecture (Overview) 10-10
Primary Database Processes 10-11
Standby Database Processes 10-12
Physical Standby Database: Redo Apply Architecture 10-13
Logical Standby Database: SQL Apply Architecture 10-14
Automatic Gap Detection and Resolution 10-15
Data Protection Modes 10-16
Data Guard Operational Requirements: Hardware and Operating System 10-18
Data Guard Operational Requirements: Oracle Database Software 10-19

Benefits of Implementing Oracle Data Guard 10-20
Summary 10-21

11 Creating a Physical Standby Database by Using SQL and RMAN Commands

Objectives 11-2
Steps to Create a Physical Standby Database 11-3
Preparing the Primary Database 11-4
FORCE LOGGING Mode 11-5
Configuring Standby Redo Logs 11-7
Creating Standby Redo Logs 11-8
Using SQL to Create Standby Redo Logs 11-9
Viewing Standby Redo Log Information 11-10
Setting Initialization Parameters on the Primary Database to Control Redo Transport 11-11
Setting LOG_ARCHIVE_CONFIG 11-12
Setting LOG_ARCHIVE_DEST_n 11-14
Specifying Role-Based Destinations 11-15
Combinations for VALID_FOR 11-17
Defining the Redo Transport Mode 11-18
Setting Initialization Parameters on the Primary Database 11-19
Specifying Values for DB_FILE_NAME_CONVERT 11-20
Specifying Values for LOG_FILE_NAME_CONVERT 11-21
Specifying a Value for STANDBY_FILE_MANAGEMENT 11-22
Example: Setting Initialization Parameters on the Primary Database 11-23
Creating an Oracle Net Service Name for Your Physical Standby Database 11-24
Creating an Entry for Your Standby Database for the Listener 11-25
Copying Your Primary Database Password File to the Physical Standby Database Host 11-26
Creating an Initialization Parameter File for the Physical Standby Database 11-27
Creating Directories for the Physical Standby Database 11-28
Starting the Physical Standby Database 11-29
Setting FAL_CLIENT and FAL_SERVER Initialization Parameters 11-30
Creating an RMAN Script to Create the Physical Standby Database 11-31
Creating the Physical Standby Database 11-33
Enabling Real-Time Apply 11-34
Starting Redo Apply 11-36
Special Note: Standby Database on the Same System 11-37
Preventing Primary Database Data Corruption from Affecting the Standby Database 11-38
Summary 11-40

12 Oracle Data Guard Broker: Overview

- Objectives 12-2
- Oracle Data Guard Broker: Features 12-3
- Data Guard Broker: Components 12-4
- Data Guard Broker: Configurations 12-5
- Data Guard Broker: Management Model 12-6
- Data Guard Broker: Architecture 12-7
- Data Guard Monitor: DMON Process 12-8
- Benefits of Using the Data Guard Broker 12-9
- Comparing Configuration Management with and Without the Data Guard Broker 12-10
- Data Guard Broker Interfaces 12-11
- Using the Command-Line Interface of the Data Guard Broker 12-12
- Using Oracle Enterprise Manager 10g Grid Control 12-14
- Data Guard Overview Page 12-15
- Benefits of Using Enterprise Manager 12-16
- Summary 12-17

13 Configuring Data Protection Modes

- Objectives 13-2
- Data Protection Modes and Redo Transport Modes 13-3
- Data Protection Modes 13-4
 - Maximum Protection Mode 13-5
 - Maximum Availability Mode 13-6
 - Maximum Performance Mode 13-7
- Comparing Data Protection Modes 13-8
- Setting the Data Protection Mode by Using DGMGRL 13-9
- Setting the Data Protection Mode 13-10
- Summary 13-12

14 Grid Infrastructure Installation

- Objectives 14-2
- Module 1: Grid Infrastructure Preinstallation Tasks 14-3
- Shared Storage Planning for Grid Infrastructure 14-4
- Sizing Shared Storage for Oracle Clusterware 14-5
- Managing Voting Disks in ASM 14-6
- Oracle Grid Infrastructure 11g Installation 14-7
- Checking System Requirements 14-8
- Enabling the Name Service Cache Daemon (nscd) 14-9
- Single-Client Access Name for the Cluster 14-10
- Checking Network Requirements 14-11

IP Address Requirements with GNS	14-13
IP Address Requirements for Manual Configuration	14-14
Broadcast and Multicast Requirements	14-16
Interconnect NIC Guidelines	14-17
Redundant Interconnect Usage	14-18
Interconnect Link Aggregation: Single Switch	14-19
Interconnect Link Aggregation: Multiswitch	14-21
Additional Interconnect Guidelines	14-22
Software Requirements (Kernel)	14-23
32-Bit Software Requirements: Packages	14-24
64-Bit Software Requirements: Packages	14-25
Oracle Validated Configuration RPM	14-26
Oracle Pre-Install RPM	14-28
Creating Groups and Users	14-29
Creating Groups and Users: Example	14-30
Shell Settings for the Grid Infrastructure User	14-31
Module 2: Grid Infrastructure Installation	14-33
Installing Grid Infrastructure	14-34
Choosing an Installation Type	14-35
Grid Plug and Play Support	14-36
Cluster Node Information	14-37
Specify Network Interface Usage	14-38
Storage Option Information	14-39
Specify Cluster Configuration: Typical Installation	14-40
Install Locations: Typical Installation	14-42
Failure Isolation Support with IPMI	14-43
Privileged Operating System Groups	14-45
Installation and Inventory Locations	14-46
Prerequisite Checks	14-47
Finishing the Installation	14-48
Verifying the Grid Infrastructure Installation	14-50
Modifying Oracle Clusterware Binaries After Installation	14-51
Module 3: Configuring ASM Disk Groups	14-53
Creating a New Disk Group	14-54
Creating a New Disk Group with ASMCMD	14-56
Creating an ASM Disk Group with ASMCA	14-57
Creating an ASM Disk Group: Advanced Options	14-58
Creating a Disk Group with Enterprise Manager	14-59
Summary	14-61

15 Administering Oracle Clusterware

- Objectives 15-2
- Managing Oracle Clusterware 15-3
- Controlling Oracle Clusterware 15-4
- Verifying the Status of Oracle Clusterware 15-5
- Determining the Location of Oracle Clusterware Configuration Files 15-6
- Checking the Integrity of Oracle Clusterware Configuration Files 15-7
- Backing Up and Recovering the Voting Disk 15-8
- Adding, Deleting, or Migrating Voting Disks 15-9
- Locating the OCR Automatic Backups 15-10
- Changing the Automatic OCR Backup Location 15-11
- Adding, Replacing, and Repairing OCR Locations 15-12
- Removing an Oracle Cluster Registry Location 15-13
- Migrating OCR Locations to ASM 15-14
- Migrating OCR from ASM to Other Shared Storage 15-15
- Performing Manual OCR Backups 15-16
- Recovering the OCR by Using Physical Backups 15-17
- Recovering the OCR by Using Logical Backups 15-18
- Oracle Local Registry 15-19
- Summary 15-21

16 Real Application Clusters Database Installation

- Objectives 16-2
- Installing the Oracle Database Software 16-3
- Creating the Cluster Database 16-8
- Database Type Selection 16-9
- Database Identification 16-10
- Cluster Database Management Options 16-11
- Passwords for Database Schema Owners 16-12
- Database File Locations 16-13
- Recovery Configuration 16-14
- Database Content 16-15
- Initialization Parameters 16-16
- Database Storage Options 16-17
- Create the Database 16-18
- Monitoring Progress 16-19
- Postinstallation Tasks 16-20
- Summary 16-21

1

Configuring for Recoverability

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

- Invoke RMAN and set and list simple configurations
- Configure your database in ARCHIVELOG mode
- Configure multiple archive log file destinations to increase availability
- Specify a retention policy
- Configure the Flash Recovery Area
- Describe the benefits of using the Flash Recovery Area



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Purpose of Backup and Recovery Functionality

The backup and recovery functionality is needed for the following:

- Data protection
 - Media failure
 - User errors
 - Application errors
- Data preservation
- Data transfer



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You need to be able to recover when there are problems introduced into your database. When you take backups of your database, you protect against problems such as media failure, user errors, and application errors. Media errors cause data problems by failing at the hardware level; a bad controller or disk drive can introduce either subtle or obvious errors. Users can also cause data errors, simply by issuing commands that should not be issued. Those same types of errors can be caused by an application with a bug.

Backup also provides for the preservation of data. You may want to create a copy of the database as of a specifically meaningful point in time and store it for a long term. This could be for possible future restoration, or it might be simply to meet compliance regulations.

You can also use backup and recovery tools to move data to other databases—even in other locations. A backup of the database is an efficient way to do this; back up the database and restore it at another location.

Typical Backup and Recovery Tasks

To be able to recover from data loss problems with minimal down time, you should be prepared to do the following:

- Configure the database for recoverability.
- Define a backup schedule.
- Plan and test different types of failure scenarios.
- Monitor and troubleshoot the backup and recovery environment.
- Restore data from backups.
- Recover transactions to a desired point in time.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A robust backup and recovery plan is important for a database that you cannot afford to lose. The plan includes the following tasks:

- **Configuration:** The backup and recovery environment needs to be configured for your environment. This includes setting, for example, the destination for backups, how old a backup should be before it is deleted, and encryption.
- **Scheduling:** Backups can be scheduled so that you do not have to manually start them. This is useful, because the best time to take a backup is often during off hours.
- **Testing:** You should plan and carry out test scenarios where data is damaged or lost, and you use a backup you have taken to recover. This should be a regular practice so you know you are successfully backing up what you need to.
- **Monitoring:** Performing backups require resources and this can affect other operations in the database. You should monitor backup and recovery tasks and ensure that they are running efficiently.
- **Restoration:** When you need to rely on a backup, you have to restore data from it. This brings the files into the database, and usually puts the state of the database into a point in the past.

- **Recovery:** If, after restoring files from a backup, you need to bring the database back into a point in time closer (or equal to) the present, then you need to perform recovery. This is the process of applying redo data to the restored data.

Oracle Backup and Recovery Solutions

These Oracle utilities and features provide the tools necessary to maintain a recoverable system:

- Recovery Manager (RMAN)
 - Incremental backups
 - Block media recovery
 - Unused block compression
 - Binary compression
 - Backup encryption
- Data Pump



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following are major backup and recovery solutions:

- **Recovery Manager:** A command-line tool for performing backup and recovery. Some of the major features available when using RMAN are:
 - **Incremental backups:** A type of backup in which only data blocks that have changed since the last incremental backup are written to the backup
 - **Block media recovery:** A method of recovering specific blocks of data, as opposed to entire tables (with Data Pump) or data files (with RMAN)
 - **Unused block compression:** A space-saving method by which blocks that have never been used are not written to the backup
 - **Binary compression:** A space-saving feature in which backup files are compressed using well-known algorithms (comparable to utilities such as `zip` on Linux)
 - **Backup encryption:** A security device for protecting backups you make
- **Data Pump:** A command-line tool for exporting and importing table data from and to operating system (OS) files

Using Recovery Manager

```
$ rman target /  
  
RMAN> BACKUP DATABASE;  
Starting backup at 10-JUN-07  
. . .  
RMAN> LIST BACKUP;  
BS Key Type LV Size Device Type Elapsed Time Completion Time  
-----  
1 Full 1.06G DISK 00:01:49 10-JUN-07  
. . .  
RMAN> DELETE OBSOLETE;  
. . .  
Do you really want to delete the above objects (enter YES or NO)? YES  
deleted archived log  
. . .
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Invoke RMAN at the operating system command line, and optionally provide command-line options.

The following are the most commonly used options:

- **target**: The connect string for the target database
- **catalog**: The connect string for a recovery catalog
- **nocatalog**: Specifies there is no recovery catalog. This is the default.
- **cmdfile**: The name of an input command file
- **log**: The name of the output message log file

The RMAN invocation shown in the slide simply connects to the local database, as the target.

Here is an example of an RMAN invocation that connects to the local database using OS authentication, and specifies a command file to be run and a log file to receive a transcript of the RMAN commands that belong to the session:

```
$ rman target / cmdfile=~/fullbu.rman log=~/fullbu.log
```

At the RMAN prompt, you can submit RMAN commands to manage your backup environment and create backups in many different ways, depending on your needs. The slide shows the commands to list the existing backups (LIST BACKUP) and delete any obsolete backups (DELETE OBSOLETE).

The specifics of what these and other commands do are covered throughout this course.

Note: Refer to the *Oracle Database Backup and Recovery User's Guide* for more information about how to invoke RMAN. Refer to the *Oracle Database Backup and Recovery Reference* for the complete list of RMAN commands and their options.

Types of RMAN Commands

RMAN commands are of the following types:

- Stand-alone command:
 - Is executed individually at the RMAN prompt
 - Cannot appear as subcommands within RUN
- Job command:
 - Must be within the braces of a RUN command
 - Is executed as a group

Some commands can be executed as either a stand-alone or a job command.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can issue two basic types of RMAN commands: stand-alone and job commands.

Stand-alone commands are executed at the RMAN prompt and are generally self-contained. Some of the stand-alone commands are:

- CHANGE
- CONNECT
- CREATE CATALOG, RESYNC CATALOG
- CREATE SCRIPT, DELETE SCRIPT, REPLACE SCRIPT

Job commands are usually grouped and executed sequentially inside a command block. If any command within the block fails, RMAN ceases processing; no further commands within the block are executed. The effects of any already executed commands still remain, though; they are not undone in any way.

An example of a command that can be run only as a job command is ALLOCATE CHANNEL. The channel is allocated only for the execution of the job, so it cannot be issued as a stand-alone command. There are some commands that can be issued either at the prompt or within a RUN command block, such as BACKUP DATABASE. If you issue stand-alone commands, RMAN allocates any needed channels by using the automatic channel allocation feature.

You can execute stand-alone and job commands in interactive mode or batch mode.

Job Commands: Example

Job commands appear inside a RUN command block:

```
RMAN> RUN
2> {
3>   ALLOCATE CHANNEL c1 DEVICE TYPE DISK
4>     FORMAT "/disk2/%U";
5>   BACKUP AS BACKUPSET DATABASE;
6>   SQL 'alter system archive log current';
7> }
```

Execution of entire block starts
when this line is entered.

Deallocated after the
RUN block completes



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

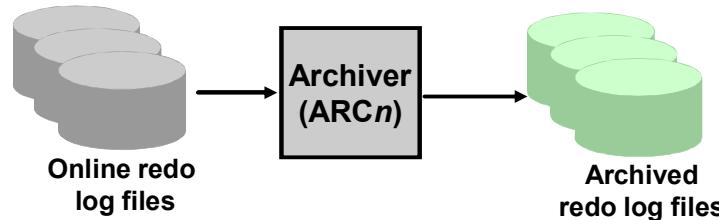
Unlike stand-alone commands, job commands must appear within the braces of a RUN command. Commands placed inside a RUN block as shown in the slide are run as a single unit of commands. Any configurations made within the run block apply within the scope of the block and override any previously made settings. The following are examples of job commands, which must appear inside a RUN block:

- ALLOCATE CHANNEL
- SWITCH

RMAN executes the job commands inside a RUN command block sequentially. If any command within the block fails, then RMAN ceases processing; no further commands within the block are executed. In effect, the RUN command defines a unit of command execution. When the last command within a RUN block completes, the Oracle Database releases any server-side resources such as input/output (I/O) buffers or I/O slave processes allocated within the block.

Configuring Your Database for Backup and Recovery Operations

- Operate the database in ARCHIVELOG mode.



- Configure the Flash Recovery Area.



ORACLE®

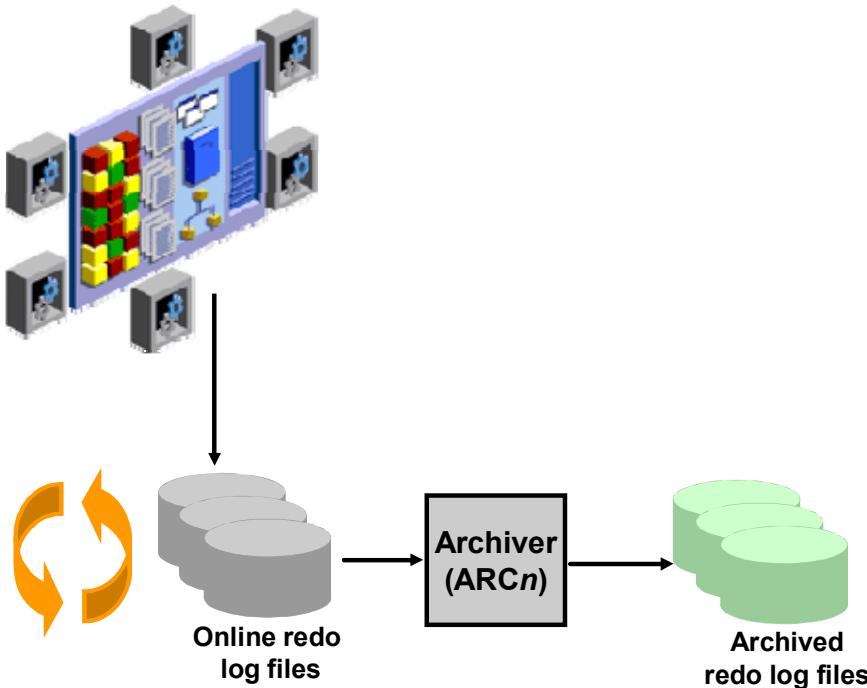
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When you operate your database in ARCHIVELOG mode, you have more recovery options after a data loss, including point-in-time recovery of the database or some tablespaces.

It is recommended that you take advantage of the Flash Recovery Area to store as many backup and recovery-related files as possible, including disk backups and archived redo logs.

Some features of Oracle Database backup and recovery, such as Oracle Flashback Database and guaranteed restore points, require the use of a Flash Recovery Area.

ARCHIVELOG Mode



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

As modifications to data in the database are made, the redo data is written out to the online redo log file. A given file is specified as being written to at a given time. When it is full, the Archiver process (ARCn) copies the online log file to another location that serves as an archive of that file, which can be preserved for as long as you need it. This provides more opportunities for recovery, because you can save, back up, and restore all of the archive redo logs ever generated.

Because the online redo log files are reused in a circular fashion, there is a protocol for controlling when one is allowed to be reused. In ARCHIVELOG mode, the database begins writing to an online redo log file only if it has been archived. This ensures that every redo log file has a chance to be archived.

Configuring ARCHIVELOG Mode

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

- Using Enterprise Manager
 1. Select the ARCHIVELOG Mode check box.
 2. Click Apply. The database can be set to ARCHIVELOG mode only from the MOUNT state.
 3. Click Yes when asked whether you want to restart the database.
- Using SQL commands
 1. Mount the database.
 2. Issue the ALTER DATABASE ARCHIVELOG command.
 3. Open the database.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Placing the database in ARCHIVELOG mode prevents redo logs from being overwritten until they have been archived.

In Enterprise Manager, do this by navigating to Availability > Recovery Settings and selecting the ARCHIVELOG Mode check box. The database must be restarted after making this change.

To issue the SQL command to put the database in ARCHIVELOG mode, the database must be in MOUNT mode. To get to the MOUNT state, the database must be in the SHUTDOWN state; if the database is currently open, you must shut it down, and then mount it. The following shows the commands to shut down an open database, put it in ARCHIVELOG mode, and then open it:

```
SQL> SHUTDOWN IMMEDIATE
SQL> STARTUP MOUNT
SQL> ALTER DATABASE ARCHIVELOG;
SQL> ALTER DATABASE OPEN;
```

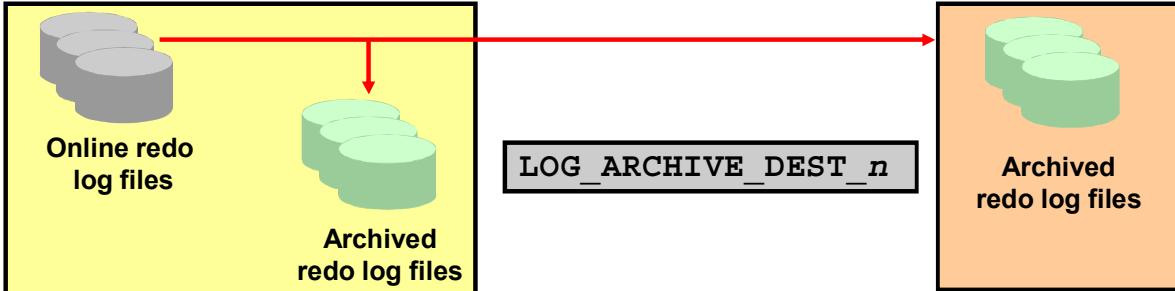
With the database in NOARCHIVELOG mode (the default), recovery is possible only until the time of the last backup. All transactions made after that backup are lost.

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

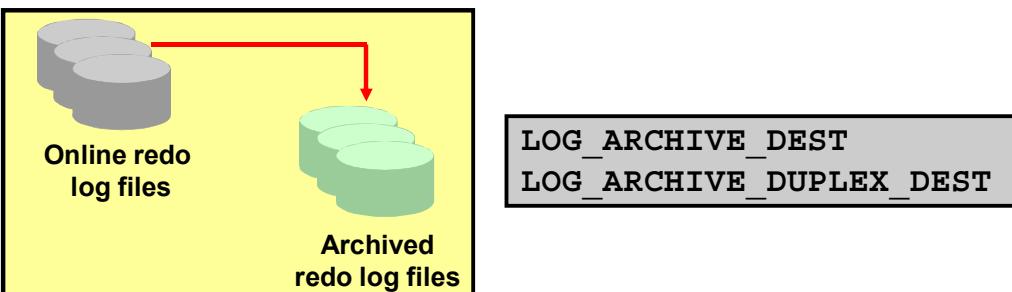
Note: Back up your database after switching to ARCHIVELOG mode because your database is recoverable only from the first backup taken in that mode.

Configuring Archive Log Destinations

- Local and remote destinations:



- Local-only destinations:



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

There are two models you can choose from, for specifying where archive log files are to be written:

- **Local and remote destinations:** Specify local and remote destinations by setting the set of `LOG_ARCHIVE_DEST_n` initialization parameters. There are ten of these, so `n` can be 1 through 10. To specify a local storage location, supply a local directory name for the value of one of these variables, supplying the `"LOCATION=` string. For example, to specify the `/disk3/arch` directory, set one of these variables as follows:

```
LOG_ARCHIVE_DEST_1 = 'LOCATION=/disk3/arch'
```

If you want to specify a remote location for a standby database, use the `SERVICE` keyword in the value, as in the following example, where `standby1` is the network service name for the standby database instance:

```
LOG_ARCHIVE_DEST_2 = 'SERVICE=standby1'
```

- **Local-only destinations:** Another option for specifying destinations supports only local disk locations. Set the `LOG_ARCHIVE_DEST` and the `LOG_ARCHIVE DUPLEX_DEST` parameters to local disk directories. Thus, you can have up to two archive log file locations. For example:

```
LOG_ARCHIVE_DEST = '/disk1/arch'
```

```
LOG_ARCHIVE DUPLEX_DEST = '/disk2/arch'
```

Oracle recommends that you use the `LOG_ARCHIVE_DEST_n` method, because this allows the most flexibility in type of destinations, and also number of destinations.

Guaranteeing Archive Log Success



```
LOG_ARCHIVE_MIN_SUCCEED_DEST = 2
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you have specified more than one destination for archive log files, you should specify a minimum number of them that are required to succeed in order for the archive to be considered successful. Do this by using the `LOG_ARCHIVE_MIN_SUCCEED_DEST` initialization parameter. Set it to the number of destinations that must succeed in receiving the archived log file. The online log file is not reused until this number is met.

In the example in the slide, there are three destinations specified: two are local, and one is remote. `LOG_ARCHIVE_MIN_SUCCEED_DEST` is set to 2, which means as long as at least two of the destinations succeed, the online redo log file can be overwritten. The example shows that destination 1 has failed. That does not stop the database, because two of them succeeded.

You can use this parameter with either of the models described in the previous slide. If you use it with the `LOG_ARCHIVE_DEST_n` model, this parameter can have values ranging from 1 through 10. If you use it with the `LOG_ARCHIVE_DEST` model, the values can be 1 or 2, because you can specify only two destinations in that case.

Specifying MANDATORY and OPTIONAL

When you define a destination, you can specify that it is a mandatory one. Do this by specifying the MANDATORY or OPTIONAL keyword after the location specification. Here is an example:

```
LOG_ARCHIVE_DEST_1 = 'LOCATION=/disk3/arch MANDATORY'
```

The default is OPTIONAL.

A mandatory destination is given special consideration. If any mandatory destination fails, Oracle Database considers the archiving of the log to have not succeeded, and the online redo log file is not allowed to be overwritten. In this case, it ignores the LOG_ARCHIVE_MIN_SUCCCEED_DEST parameter.

Any destination specified by LOG_ARCHIVE_DEST is mandatory. Any destination declared by LOG_ARCHIVE_DUPLEX_DEST is optional if LOG_ARCHIVE_MIN_SUCCCEED_DEST = 1 and mandatory if LOG_ARCHIVE_MIN_SUCCCEED_DEST = 2.

Specifying a Backup Destination

Backups can be written to:

- Disk directory
- Tape, using Oracle Secure Backup
- Media Management Library
 - Tape
 - Disk or tape, using proxy copy
- Flash Recovery Area: Disk area set aside for backup and recovery and flashback database purposes



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Backups can be written to a designated disk directory, a Media Management Library (MML), or the Flash Recovery Area. Specifying a disk directory or the Flash Recovery Area means that backups go to hard-disk media. Typically, backups are regularly moved offline to tape via the media management interface in order to maintain disk space availability. Any disk directory can be specified as the destination of a backup provided that it already exists. A media management library can be used to copy files to tape devices, or to carry out proxy copies. A proxy copy is where the MML is requested to make a copy of a file to a disk or tape device. The MML must be able to provide the proxy copy service for this to work.

If you set up a Flash Recovery Area, many backup and recovery tasks are simplified for you. The Oracle Database automatically names files for you, and deletes obsolete files when there is space pressure. More information about configuring a Flash Recovery Area is provided later in this lesson.

To specify that backups are to be written to disk, use this command:

```
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK;
```

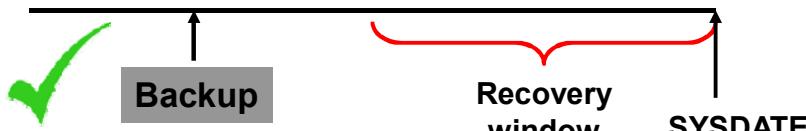
Subsequently, when backups are made, if the FORMAT keyword is used (that specifies a disk directory location for the backup), then the backup is written there. If there is a Flash Recovery Area configured, then it goes there; otherwise, backups are written to a platform-specific default location.

To specify that a tape device is to be used, use this command:

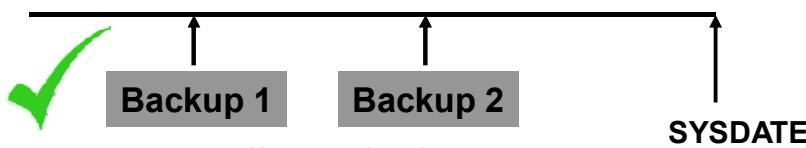
```
RMAN> CONFIGURE DEFAULT DEVICE
```

Specifying a Retention Policy

- Retention policy: Describes which backups will be kept and for how long
- Two types of retention policies:
 - **Recovery window:** Establishes a period of time within which point-in-time recovery must be possible



- **Redundancy:** Establishes a fixed number of backups that must be kept



- Retention policies are mutually exclusive.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A *retention policy* describes which backups will be kept and for how long. You can set the value of the retention policy by using the RMAN CONFIGURE command or Enterprise Manager.

Recovery Window Retention Policy

The best practice is to establish a period of time during which it will be possible to discover logical errors and fix the affected objects by doing a point-in-time recovery to just before the error occurred. This period of time is called the *recovery window*. This policy is specified in number of days. For each data file, there must always exist at least one backup that satisfies the following condition:

SYSDATE - backup_checkpoint_time >= recovery_window

You can use the following command syntax to configure a recovery window retention policy:

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF <days>
      DAYS;
```

where <days> is the size of the recovery window.

If you are not using a recovery catalog, you should keep the recovery window time period less than or equal to the value of the control file parameter `CONTROL_FILE_RECORD_KEEP_TIME` to prevent the record of older backups from being overwritten in the control file. If you are using a recovery catalog, make sure that `CONTROL_FILE_RECORD_KEEP_TIME` is greater than the time period between catalog resynchronizations. Resynchronizations happen when you:

- Create a backup. In this case, the synchronization is done implicitly.
- Execute the `RESYNC CATALOG` command

Redundancy Retention Policy

If you require a certain number of backups to be retained, you can set the retention policy on the basis of the redundancy option. This option requires that a specified number of backups be cataloged before any backup is identified as obsolete. The default retention policy has a redundancy of 1, which means that only one backup of a file must exist at any given time. A backup is deemed obsolete when a more recent version of the same file has been backed up.

You can use the following command to reconfigure a redundancy retention policy:

```
RMAN> CONFIGURE RETENTION POLICY TO REDUNDANCY <copies>;
```

where <copies> is the number of copies that are required for policy satisfaction.

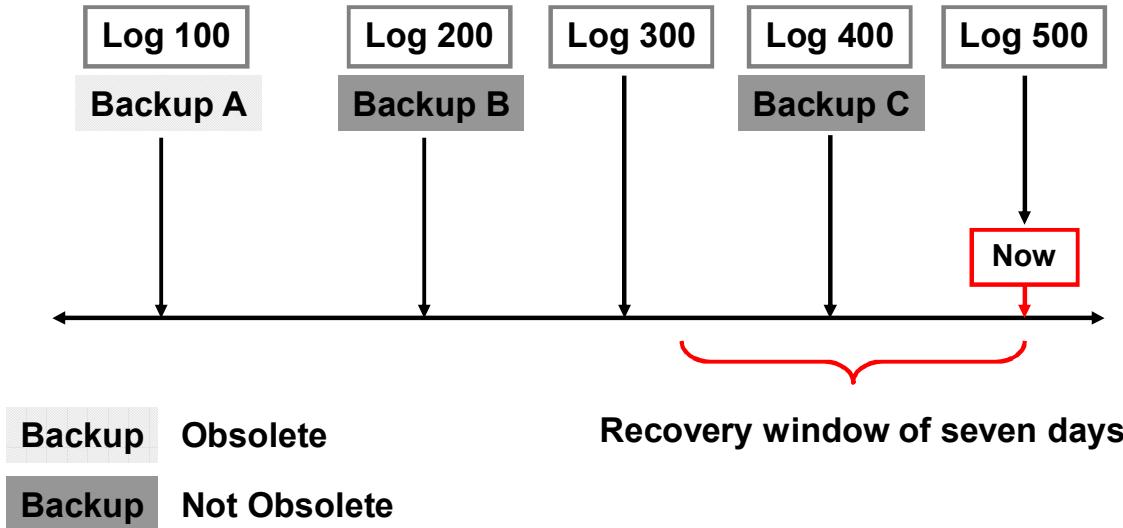
Disabling the Retention Policy

You may want to disable the retention policy totally. If you have a separate system, outside of RMAN, that backs up your disk backups to tape, you may want to do this. If you disable the retention policy, RMAN never considers a backup obsolete. Because RMAN does not have to decide when to remove a backup from disk (because another utility is managing that), RMAN does not need to be configured for making that decision. In this case, records of each backup are maintained for as long as is specified by the `CONTROL_FILE_RECORD_KEEP_TIME` initialization parameter. Disable the retention policy by using this command:

```
RMAN> CONFIGURE RETENTION POLICY TO NONE;
```

Note: You can specify that a backup is an exception to the retention policy that you have defined. This is called an archival backup, which is covered in the lesson titled “Using RMAN to Create Backups.”

A Recovery Window Retention Policy: Example



Backup B and archive logs 201 through 500 are required to satisfy this retention policy.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

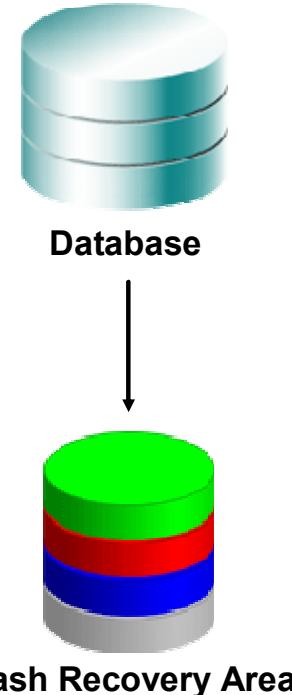
The retention policy in the slide shows that it requires the ability to recover to any time within the last seven days. Some of the backups and logs are obsolete, because they are not needed to recover to a time within the seven-day window. This retention policy is configured thus:

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 7 DAYS;
```

Given the backups and archived log files available, the only data needed to recover to a point inside the recovery window is Backup B and logs 201 through 500. Note that Backup A is not needed because there is a later backup (B) that is still before the recovery window. Also, Backup C is not sufficient as the only backup to retain because it would not satisfy a need to recover to points in time at the beginning of the recovery window. The last backup that was taken before the beginning of the recovery window, including all logs since that backup, is what is necessary.

Using a Flash Recovery Area

- Permanent items:
 - Multiplexed copies of the current control file
 - Multiplexed copies of online redo logs
- Transient items:
 - Archived redo logs
 - Data file copies
 - Control file copies
 - Control file autobackups
 - Backup pieces
 - Flashback logs



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Flash Recovery Area is a unified storage location for all recovery-related files and activities in an Oracle Database. All files that are needed to completely recover a database from a media failure are part of the Flash Recovery Area. The recovery-related files are of two types: permanent and transient. Permanent files are actively being used by the instance. Transient files are needed only in the event of some type of recovery operation.

Permanent Items

- **Control file:** Depending on the setting of several initialization parameters, a copy of the control file is created in the Flash Recovery Area location when you create a new database or control file. For details, see the “Semantics” section of the CREATE CONTROLFILE command in the *Oracle Database SQL Language Reference*.
- **Multiplexed copies of online redo log files:** A mirrored copy from each redo log group can be here. When you create a database, you can specify the location of the online redo log files using the LOGFILE clause. If you do not include that clause, the locations are set according to the values of the following initialization parameters:
 - **DB_CREATE_ONLINE_LOG_DEST_n:** If one or more of these variables are set, these are the only locations used.
 - **DB_CREATE_FILE_DEST:** If this is set, this is the primary file location.
 - **DB_RECOVERY_FILE_DEST:** If this is set, in addition to DB_CREATE_FILE_DEST, then this location is used as the mirror.

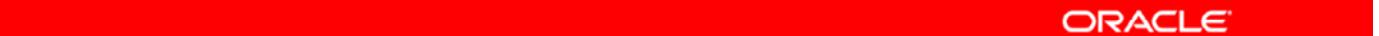
For more details on how these variables affect the location of the online redo logs, see the `LOGFILE` clause of the `CREATE DATABASE` statement in the *Oracle Database SQL Language Reference*.

Transient Items

- **Archived redo log files:** When the Flash Recovery Area is configured, `LOG_ARCHIVE_DEST_10` is automatically set to the Flash Recovery Area location. The Archiver background process creates archived redo log files in the Flash Recovery Area and in other configured `LOG_ARCHIVE_DEST_n` locations. If no `LOG_ARCHIVE_DEST_n` locations are defined, the default location for archived redo log files is in the Flash Recovery Area.
- **Flashback logs:** Flashback logs are generated when Flashback Database is enabled.
- **Control file autobackups:** The default location for control file autobackups created by RMAN and autobackups generated by the Oracle Database server is the Flash Recovery Area.
- **Data file copies:** The `BACKUP AS COPY` command creates image data file copies in the Flash Recovery Area.
- **RMAN files:** The Flash Recovery Area is the default location that is used by RMAN for backups and restoration of the archive log content from tape for a recovery operation.

Defining a Flash Recovery Area

- The Flash Recovery Area is defined by setting both of the following initialization parameters:
 - **DB_RECOVERY_FILE_DEST_SIZE**: Sets the disk limit
 - **DB_RECOVERY_FILE_DEST**: Sets the location for the Flash Recovery Area
- These parameters are dynamic.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the following mandatory parameters to define the Flash Recovery Area:

- **DB_RECOVERY_FILE_DEST_SIZE**: You must define a disk limit, which is the amount of space that the Flash Recovery Area is permitted to use. Setting a limit allows the remaining disk space not dedicated to the Flash Recovery Area to be used for other purposes. A basic recommendation for the size of the disk limit is the sum of the database size, the size of incremental backups, and the size of all archive log files that have not been copied to tape. The minimum size of the Flash Recovery Area should be at least large enough to contain archived redo log files that have not been copied to tape. The sizing of the Flash Recovery Area is dependent on the backup strategy and other options that are implemented. Flashback database set points, and multilevel incremental backups all have an effect on the size of the Flash Recovery Area.
- **DB_RECOVERY_FILE_DEST**: A Flash Recovery Area specification contains a location, which is a valid destination to create files.

Defining a Flash Recovery Area Using Enterprise Manager

Flash Recovery

Flash Recovery Area is enabled for this database. The chart shows space used by each file type that is not reclaimable by Oracle. Performing backups to a 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 Flash Recovery Area Size must be set when the location is set

Reclaimable Flash Recovery Area (B) **0**

Free Flash Recovery Area **2.62** (GB)

Enable Flashback Database - flashback logging can be used for fast database point-in-time recovery*

The flashback area must be set to enable flashback logging. When using flashback logs, you may recover your entire 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.

Flashback Retention Time n/a

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

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

Flashback Time **n/a**

Apply changes to SPFILE only. Otherwise the changes will be made to both SPFILE and the running instance which requires that you restart the database to invoke static parameters.

Flash Recovery Area Usage

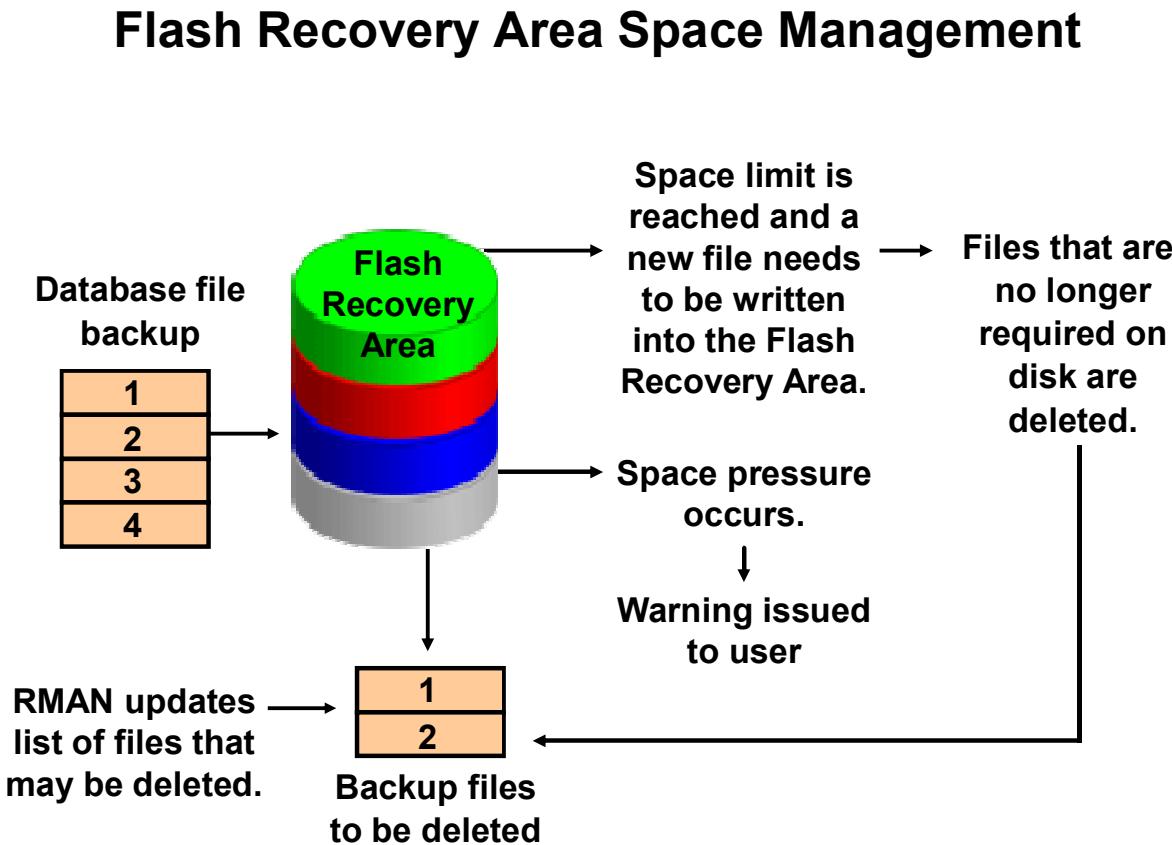
File Type	Size (GB)	Percentage
BACKUP PIECE	1.13	28.4%
REDO LOG	0.15	3.7%
ARCHIVED LOG	0.09	2.3%
CONTROL FILE	0.01	0.2%
IMAGE COPY	0	0%
FLASHBACK LOG	0	0%
Usable	2.62	65.5%

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use Enterprise Manager Grid Control and Database Control to easily define the Flash Recovery Area. On the Database Control home page, navigate to Availability > Recovery Settings in the Backup/Recovery Settings region. You can define the Flash Recovery Area location and its size on the Recovery Settings page.

You must set the size of the Flash Recovery Area when specifying its location.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Each time RMAN creates a file in the Flash Recovery Area, the list of files that are no longer required on disk is updated. Based on the value of `DB_RECOVERY_FILE_DEST_SIZE`, when the Flash Recovery Area experiences space pressure or is low on free space because there are no files that can be deleted from the Flash Recovery Area, you are warned of the danger of running out of space. The Oracle Database server and RMAN continue to create files in the Flash Recovery Area until 100% of the disk limit is reached. When setting `DB_RECOVERY_FILE_DEST_SIZE`, you must allocate enough space to hold the recovery files, including backups that are waiting to be backed up to tape. Files that are obsolete or have been backed up to tape are likely candidates for deletion to provide free space.

When a file is written into the Flash Recovery Area and space is needed for that file, the Oracle Database server deletes a file that is on the obsolete files list. When a file is written and deleted from the Flash Recovery Area, notification is written into the alert log.

Note: When the Flash Recovery Area's used space is at 85%, a warning alert is issued, and when used space is at 97%, a critical alert is issued. These are internal settings and cannot be changed. Following is a sample alert log output:

```
WARNING: db_recovery_file_dest_size of 52428800 bytes is 100.00% used, and has 0 remaining bytes available.
```

You can issue the following query to determine the action to take:

```
SQL> SELECT object_type, message_type, message_level,  
2 reason, suggested_action  
3 FROM dba_outstanding_alerts;
```

Your choice is to add additional disk space, back up files to a tertiary device, delete files from the Flash Recovery Area using RMAN, or consider changing the RMAN retention policy.

Flash Recovery Area Space Usage

- Configure the retention policy to the minimum value appropriate for your database.
- Back up the archive log files regularly and delete the files upon completion of the backup.
- Use the RMAN REPORT OBSOLETE and DELETE OBSOLETE commands to remove backups and file copies that are not required.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To avoid running out of space in the Flash Recovery Area, perform the following steps as needed or appropriate:

- Use RMAN to delete unnecessary files from the Flash Recovery Area.
- Use RMAN to take frequent backups of the Flash Recovery Area.
- Change the RMAN retention policy to retain backups for a smaller period of time.
- Change the RMAN archived log deletion policy.
- Add disk space and increase the value of the DB_RECOVERY_FILE_DEST_SIZE database initialization parameter if you frequently run out of space.

Enterprise Manager does not report the amount of space used by the Flash Recovery Area on the disk or the amount of space used in the Flash Recovery Area directory tree, but it does report the sizes of the files that RMAN believes are in the directory. So do not put any files in this area that are not managed by RMAN.

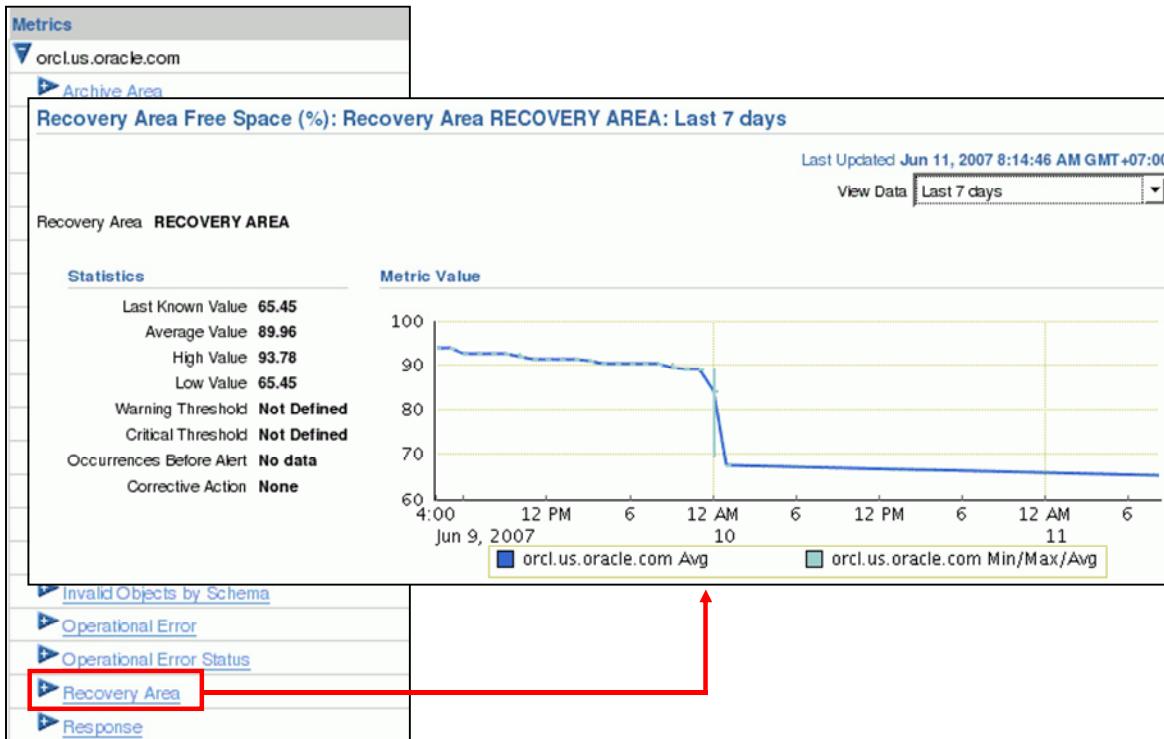
If you remove any file from this area with a tool other than RMAN, use RMAN to remove the file entries from the catalog. For example, to back up the archived log files in the Flash Recovery Area, and then delete the files after they have been successfully backed up, you would use the RMAN command as follows:

```
BACKUP ARCHIVELOG ALL DELETE ALL INPUT;
```

If you use a backup solution other than RMAN, you still have to use RMAN to remove the files from the Flash Recovery Area. After the archived redo log files have been backed up and removed from disk, use the RMAN CROSSCHECK and DELETE commands to reclaim the archived log space from the Flash Recovery Area. You should do this on a regular basis or after every backup.

You can also use the Manage Backups page of Enterprise Manager to manage backups. On that page, you can perform a cross-check operation and also delete expired and obsolete backups.

Monitoring the Flash Recovery Area



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Real-time Flash Recovery Area metrics can be viewed through Enterprise Manager Database Control. On the home page, scroll down to the Related Links section and select All Metrics. Scan the list and click Recovery Area.

The displayed page shows the Recovery Area Free Space (%) metric, which indicates the percentage of the recovery that is free space. Click the percentage number to see the graph of recovery area usage.

Benefits of Using a Flash Recovery Area

Using the Flash Recovery Area for recovery-related files:

- Simplifies the location of database backups
- Automatically manages the disk space allocated for recovery files



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Using a Flash Recovery Area for all recovery-related files simplifies the ongoing administration of your database.

Oracle Corporation recommends the use of the Flash Recovery Area for all recovery-related files.

Summary

In this lesson, you should have learned how to:

- Invoke RMAN and set and list simple configurations
- Configure your database in ARCHIVELOG mode
- Configure multiple archive log file destinations to increase availability
- Specify a retention policy
- Configure the Flash Recovery Area
- Describe the benefits of using the Flash Recovery Area



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Configuring Backup Specifications



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

- Use Enterprise Manager to configure backup settings
- Enable control file autobackup
- Allocate channels to use in backup
- Configure backup optimization



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Using RMAN to Create Backups

When creating a backup using RMAN, you can specify:

- Type: Full or incremental
- Files to back up: Entire database, data files, control file, server parameter file, archived redo log files
- Backup type: Image copy or backup set
- Proxy options: Pass on to the Media Management Library the responsibility of copying the files.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A backup is a copy of data from your database that can be used to reconstruct that data. The results of a backup created through RMAN can be either image copies or backup sets.

When performing a backup using RMAN, you can specify:

- **The type of backup to be performed:** You can perform backups of the entire database to include every used data block in the files (a **FULL** backup) or incremental backups (**INCREMENTAL**).
If **CONFIGURE CONTROLFILE AUTOBACKUP** is enabled, RMAN automatically backs up the control file and the current server parameter file after a **BACKUP** command is executed.
- **What to backup:** Valid values for a database backup are **DATABASE**, **DATAFILE**, **TABLESPACE**, **ARCHIVELOG**, **CURRENT CONTROLFILE**, and **SPFILE**. RMAN has additional commands that can be used to move backup files to tape.
- **Backup type:** Whether an image copy (**AS COPY**) or backup set (**AS BACKUPSET**) is created
 - The file name format and location for backup pieces (**FORMAT**)
 - Which data files or archived redo logs should be excluded from the backup set (**SKIP**)
 - That the input files should be deleted upon the successful creation of the backup set (**DELETE INPUT**)
- **Proxy options:** Proxy options that specify how the Media Management Library (MML) is to carry out the copying of the files. The **PROXY** option of the **BACKUP** command provides a way for you to relieve RMAN of having to know how the media that is controlled by the MML works. This option is covered in the lesson titled “Using RMAN to Create Backups.”

Backup Destinations

Backups can be written to:

- Disk directory
- Media Management Library
 - Typically used for disaster recovery, when disk backups are lost
 - Oracle Secure Backup provides one.
- Flash Recovery Area
 - This is the disk area set aside for backup and recovery and flashback database purposes.
 - Define the location and the size.
 - Files are automatically named by using Oracle Managed Files.
 - Files are automatically retained and deleted as necessary.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Backups can be written to a designated disk directory, a Media Management Library, or the Flash Recovery Area. Specifying a disk directory or the Flash Recovery Area means that backups go to hard-disk media. Typically, they are regularly moved offline to tape via the media management interface to maintain disk space availability. Any disk directory can be specified as the destination of a backup provided that it already exists.

If you configure a Flash Recovery Area, many backup and recovery tasks are simplified for you. The Oracle Database server automatically names files for you, and deletes obsolete files when there is space pressure.

Note: For more information about Oracle Secure Backup, see the *Oracle Secure Backup Administrator's Guide*.

Configuring Persistent Settings for RMAN

- RMAN is preset with default configuration settings.
- Use the `CONFIGURE` command to:
 - Configure automatic channels
 - Specify the backup retention policy
 - Specify the number of backup copies to be created
 - Set the default backup type to `BACKUPSET` or `COPY`
 - Limit the size of backup pieces
 - Exempt a tablespace from backup
 - Enable and disable backup optimization
 - Configure automatic backups of control files
 - Define the `ARCHIVELOG DELETION` policy
 - Specify the parallelism for a device
 - Set the encryption and compression parameters to be used for backups



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To simplify ongoing use of RMAN for backup and recovery, RMAN enables you to set several persistent configuration settings for each target database. These settings control many aspects of RMAN's behavior. You can save persistent configuration information, such as channel parameters, parallelism, and the default device type, in the RMAN repository. These configuration settings are always stored in the control file and in the recovery catalog database (if it exists).

These settings have default values, which allow you to use RMAN immediately. However, as you develop a more advanced backup and recovery strategy, you may have to change these settings to implement that strategy. You can use the `CONFIGURE` command to configure persistent settings for RMAN backup, restore, duplication, and maintenance jobs. These settings are in effect for any RMAN session until the configuration is cleared or changed.

Note: The configuration settings can be changed in an RMAN job (or session) just for the duration of the job (or session) with the `SET` command.

Using Enterprise Manager to Configure RMAN Settings

The screenshot shows the Oracle Enterprise Manager interface for configuring RMAN settings. At the top, it says "Database Instance: orcl.us.oracle.com" with tabs for Home, Performance, and Availability. Below that is a "Backup/Recovery" section with a "Setup" menu containing "Backup Settings" (which is highlighted with a red box), "Recovery Settings", and "Recovery Catalog Settings". The main content area is titled "Backup Settings" and contains three tabs: Device, Backup Set (which is selected and highlighted with a red box), and Policy. Under "Device", there's a "Maximum Backup Piece (File) Size" input field set to 1 MB, with a note below it: "Specify a value to restrict the size of each backup piece." Under "Backup Set", there are sections for "Tape Settings" (with fields for Copies of Datafile Backups and Copies of Archivelog Backups both set to 1) and "Host Credentials" (with fields for Username and Password). A red arrow points from the "Backup Set" tab in the main content area to the "Backup Settings" link in the navigation bar.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use Oracle Enterprise Manager to specify the backup settings for an instance. From the Database Home page, navigate to Availability > Backup Settings.

The Backup Settings property page consists of three tabs:

- **Device:** Used to set the disk and tape configuration settings, including the Media Management Library (MML) settings
- **Backup Set** (shown in the slide): Used to specify parameters for backup sets and to enter host credentials
- **Policy:** Used to set various backup and retention policies before you initiate a backup, such as automatically backing up the control file and SPFILE. The Policy page also allows you to configure block change tracking support, a feature that provides faster incremental backups.

Note: Backup settings provide the default settings for all backups taken. When creating a backup, some of these settings can be overridden for that specific backup.

Control File Autobackups

```
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

Backup Settings

Device Backup Set Policy

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 flash recovery area location.

Best practice: Oracle recommends that you enable control file autobackup.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To easily recover from the loss of all control file copies, you should configure RMAN to take automatic backups of the control file. The automatic backup of the control file occurs independently of any backup of the current control file explicitly requested as part of your backup command. If you are running RMAN in NOCATALOG mode, it is highly recommended that you activate control file autobackup. Otherwise, if you lose your control file, your database may be unrecoverable.

To configure control file autobackup, modify the backup policy for your database by using Enterprise Manager or use the following RMAN command:

```
CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

By default, control file backups are disabled. If you enable control file backups, RMAN automatically backs up the control file and the current server parameter file (if used to start up the database) in one of two circumstances:

- A successful backup is recorded in the RMAN repository.
- A structural change to the database affects the contents of the control file, which, therefore, must be backed up.

The control file autobackup file name has a default format of %F for all device types, so that RMAN can infer the file location and restore it without a repository. This variable format translates into c-IIIIIIIII-YYYYMMDD-QQ, where:

- IIIIIIIII stands for the DBID
- YYYYMMDD is a time stamp of the day the backup is generated
- QQ is the hex sequence that starts with 00 and has a maximum of FF

You can change the default format by using the CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE *type* TO '*string*' command. The value of string must contain the substitution variable %F and cannot contain other substitution variables. For example:

```
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT  
FOR DEVICE TYPE DISK TO '/u01/oradata/cf_ORCL_auto_%F';
```

Control file backups are stored in the Flash Recovery Area, unless otherwise specified.

With a control file backup, RMAN can recover the database even if the current control file, recovery catalog, and server parameter file are inaccessible. Because the path used to store the backup follows a well-known format, RMAN can search for and restore the server parameter file or control file from that backup.

Managing Persistent Settings

- Use the SHOW command to list current settings:

```
RMAN> SHOW CONTROLFILE AUTOBACKUP FORMAT;  
RMAN> SHOW EXCLUDE;  
RMAN> SHOW ALL;
```

- Use the CLEAR option of the CONFIGURE command to reset any persistent setting to its default value:

```
RMAN> CONFIGURE BACKUP OPTIMIZATION CLEAR;  
RMAN> CONFIGURE MAXSETSIZE CLEAR;  
RMAN> CONFIGURE DEFAULT DEVICE TYPE CLEAR;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Using the RMAN SHOW command, you can view the RMAN configuration settings. If SHOW ALL is executed when connected to a target database, only node-specific configurations and database configurations are displayed.

You can return to the default value for any CONFIGURE command by executing the same command with the CLEAR option.

Configuring Devices for Backup

```
RMAN> CONFIGURE DEVICE TYPE sbt PARALLELISM 3;
```

```
RMAN> CONFIGURE DEVICE TYPE DISK  
2> BACKUP TYPE TO COMPRESSED BACKUPSET;
```

```
RMAN> CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO COPY;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can configure a device to be used by RMAN using the `CONFIGURE DEVICE TYPE` command.

Parallelism

Parallelism is the number of streams of data that can be used to read from and write to the device. This effectively causes that number of channels to be allocated when the device is used by RMAN. For example, if a media manager has two tape drives available, parallelism 2 would allow both tape drives to be used simultaneously for `BACKUP` commands using that media manager. Parallelism for the disk device type is also useful, when you want to spread out a backup over multiple disks.

Specify the parallelism to be used on the device using the `PARALLELISM` clause, like this:

```
CONFIGURE DEVICE TYPE <device> PARALLELISM <n>
```

where `<n>` is the parallelism value.

Backup Type

The output of the backup can be either a backup set or an image copy. Configure the default for a device type using the `BACKUP TYPE TO` clause. Specify `BACKUPSET` for a backup set and `COPY` for an image copy.

Compression

Specify the COMPRESSED keyword after the BACKUP TYPE TO clause to specify that backups to this device are to be compressed. Compression results in smaller backup files.

Note: Compression can be applied only to backup sets.

Configuring and Allocating Channels for Use in Backups

- Configure automatic channels with the CONFIGURE command:

```
RMAN> CONFIGURE DEVICE TYPE sbt PARALLELISM 1;
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO sbt;
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt ...
RMAN> BACKUP DATABASE;
```

- Allocate channels manually with the ALLOCATE CHANNEL command within a RUN block:

```
RMAN> RUN
{
  ALLOCATE CHANNEL ch1 DEVICE TYPE DISK;
  BACKUP DATABASE PLUS ARCHIVELOG;
}
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Choose from the following options for configuring channels and executing backups:

- Configure automatic channels with the CONFIGURE command, and then issue the BACKUP command at the RMAN prompt or within a RUN block.
- Manually allocate channels with the ALLOCATE CHANNEL command within a RUN block, and then issue BACKUP commands.

Configuring Backup Optimization

- The BACKUP command skips backing up files when identical files have already been backed up.
- It is used when the following conditions are true:
 - Backup optimization is enabled.
 - BACKUP DATABASE, BACKUP ARCHIVELOG with ALL or LIKE options, or BACKUP BACKUPSET ALL commands are executed.
 - Only one type of channel is allocated.
- It can be overridden with the FORCE option.
- It is always used for RECOVERY AREA, DB_RECOVERY_FILE_DEST, and RECOVERY FILES BACKUP options.

```
RMAN> CONFIGURE BACKUP OPTIMIZATION ON;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you enable backup optimization, the BACKUP command skips backing up files when identical files have already been backed up to the specified device type.

If RMAN determines that a file is identical and it has already been backed up, then it is a candidate to be skipped. However, RMAN performs further checking to determine whether to skip the file, because both the retention policy and the backup duplexing feature are factors in the algorithm that RMAN uses to determine whether there are sufficient backups on the specified device type.

Refer to the *Oracle Database Backup and Recovery User's Guide* for detailed information about the criteria that RMAN uses to determine whether a file is identical and the backup optimization algorithm.

You can enable backup optimization on the Backup Settings page in Enterprise Manager or by issuing the CONFIGURE BACKUP OPTIMIZATION ON command. By default, backup optimization is disabled.

Backup optimization is automatically enabled for the BACKUP RECOVERY AREA | DB_RECOVERY_FILE_DEST and BACKUP RECOVERY FILES commands.

To override backup optimization and back up all files whether or not they have changed, specify the FORCE option on the BACKUP command as in the following example:

```
BACKUP DEVICE TYPE sbt BACKUPSET ALL FORCE;
```

Note that the FORCE option does not apply to files in the recovery area.

You can disable backup optimization on a persistent basis using Enterprise Manager or by issuing the following command:

```
CONFIGURE BACKUP OPTIMIZATION OFF;
```

Summary

In this lesson, you should have learned how to:

- Use Enterprise Manager to configure backup settings
- Enable control file autobackup
- Allocate channels to use in backup
- Configure backup optimization



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Using RMAN to Create Backups

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

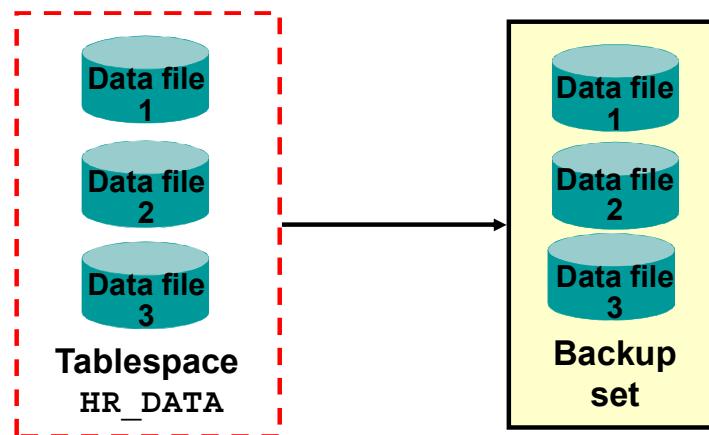
- Create image file backups
- Create a whole database backup
- Create a full database backup
- Enable fast incremental backups
- Create duplex backup sets
- Back up a backup set
- Create an archival backup for long-term retention
- Create a multisession backup
- Create a compressed backup
- Create an encrypted backup
- Report on and maintain backups



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Creating Backup Sets

```
RMAN> BACKUP AS BACKUPSET  
2> FORMAT '/BACKUP/df_%d_%s_%p.bus'  
3> TABLESPACE hr_data;
```



ORACLE

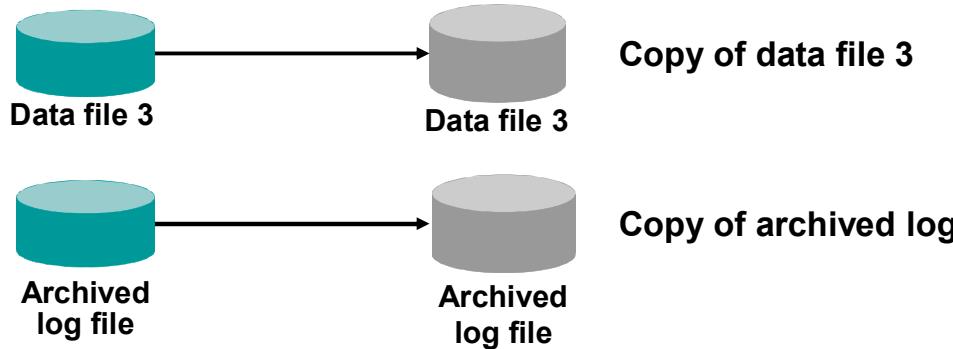
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

RMAN can store its backups in an RMAN-exclusive format called a backup set. A backup set is a collection of files called backup pieces, each of which may contain a backup of one or more database files.

Note: The `FORMAT` parameter specifies a pattern to use in creating a file name for the backup pieces created by this command. The `FORMAT` specification can also be provided through the `ALLOCATE CHANNEL` and `CONFIGURE` commands.

Creating Image Copies

```
RMAN> BACKUP AS COPY DATAFILE '/ORADATA/users_01_db01.dbf';
RMAN> BACKUP AS COPY ARCHIVELOG LIKE '/arch%';
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

An image copy is a clone of a single data file, an archived redo log, or a control file. An image copy can be created with the BACKUP AS COPY command or with an operating system command. When you create the image copy with the RMAN BACKUP AS COPY command, the server session validates the blocks in the file and records the copy information in the control file.

An image copy has the following characteristics:

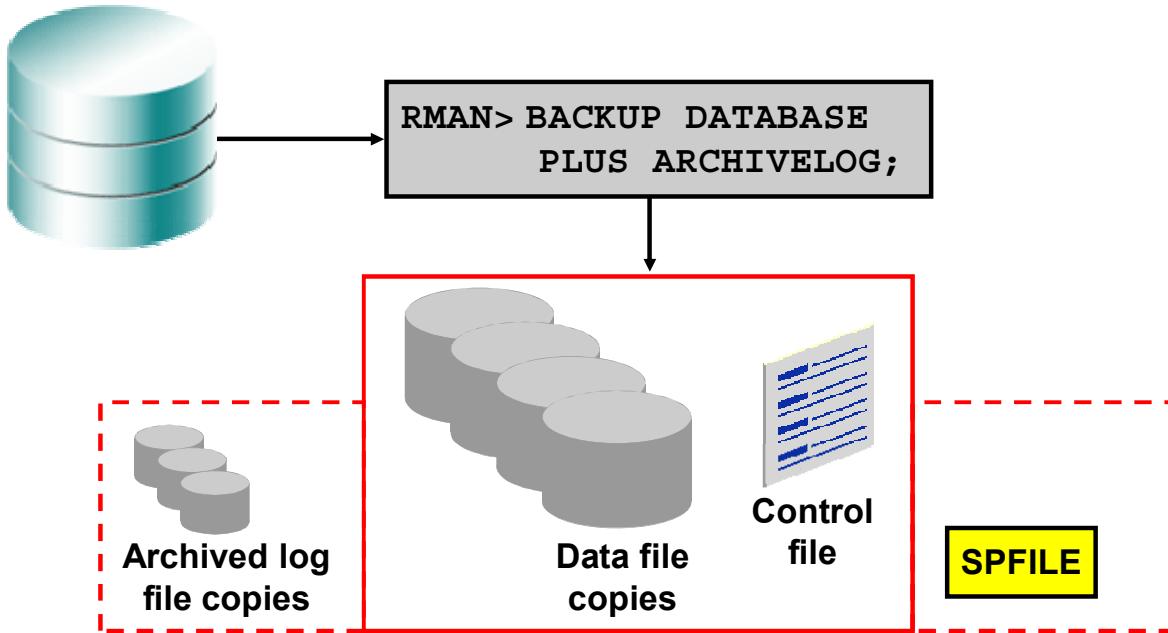
- An image copy can be written only to disk. When large files are being considered, copying may take a long time, but restoration time is reduced considerably because the copy is available on the disk.
- If files are stored on disk, they can be used immediately by using the SWITCH command in RMAN, which is equivalent to the ALTER DATABASE RENAME FILE SQL statement.
- In an image copy, all blocks are copied, whether they contain data or not, because an Oracle database process copies the file and performs additional actions such as checking for corrupt blocks and registering the copy in the control file. To speed up the process of copying, you can use the NOCHECKSUM parameter. By default, RMAN computes a checksum for each block backed up, and stores it with the backup. When the backup is restored, the checksum is verified. For more information about the NOCHECKSUM option of the BACKUP command, see the *Oracle Database Backup and Recovery Reference*.

- An image copy can be part of a full or an incremental level 0 backup because a file copy always includes all blocks. You must use the level 0 option if the copy is used in conjunction with an incremental backup set.

The example in the slide creates the following image copies:

- A copy of the /ORADATA/users01_db01.dbf data file
- A copy of the archived redo log files

Creating a Whole Database Backup



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A whole database backup can be either backup sets or image copies of the entire set of data files and must include the control file. You can optionally include the server parameter file (SPFILE) and archived redo log files. Using Recovery Manager (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 archivelog files. That causes RMAN to remove the archivelog files after backing them up. This is useful especially if you are not using a Flash 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 this:

```
RMAN> BACKUP DATABASE PLUS ARCHIVELOG DELETE INPUT;
```

You must have issued the following **CONFIGURE** commands to make the backup:

- **CONFIGURE DEFAULT DEVICE TYPE TO disk;**
- **CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO COPY;**
- **CONFIGURE CONTROLFILE AUTOBACKUP ON;**

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;
```

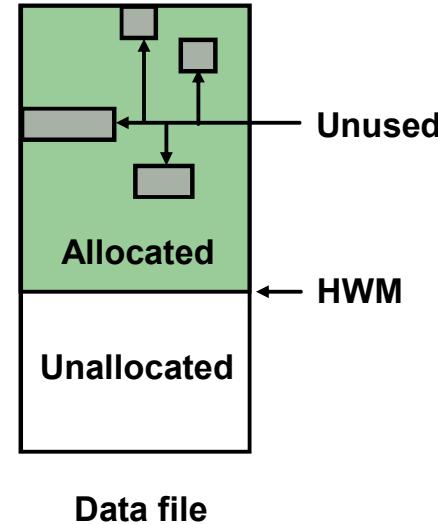
By default, RMAN executes each BACKUP command serially. However, you can parallelize the copy operation by:

- Using the CONFIGURE DEVICE TYPE DISK PARALLELISM *n* command, where *n* is the desired degree of parallelism
- Allocating multiple channels
- Specifying one BACKUP AS COPY command and listing multiple files

Saving Backup Space with Unused Block Compression

The following blocks may be skipped during certain types of backup operations:

- Unallocated blocks: These are above the data file's high-water mark (HWM).
- Unused blocks: These are blocks that have been allocated but no longer belong to a segment.



ORACLE

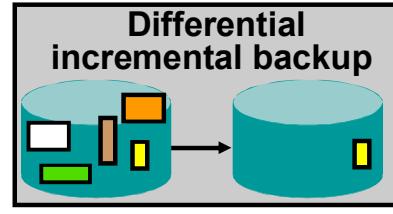
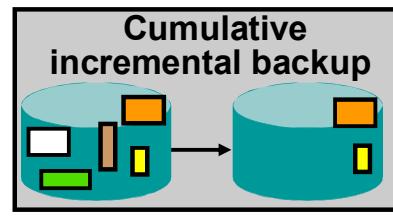
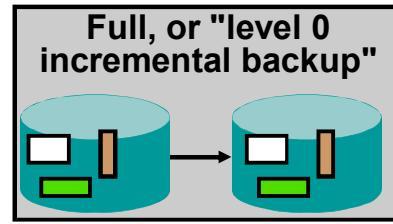
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When certain types of backups occur, RMAN is able to skip some blocks. Unallocated blocks may be skipped. They are those that have not been allocated; they are above the HWM. Also, some allocated blocks that no longer belong to a segment (are not in use) may be skipped, provided the following is true:

- There are no guaranteed restore points defined.
- The data file contains data only for locally managed tablespaces.
- The data file is being backed up to a backup set as part of a full backup or a level 0 incremental backup. These backup types are covered in the following slides.
- The backup is going to disk or Oracle Secure Backup is the media manager.

RMAN Backup Types

- A full backup contains all used data file blocks.
- A level 0 incremental backup is equivalent to a full backup that has been marked as level 0.
- A cumulative level 1 incremental backup contains only blocks modified since the last level 0 incremental backup.
- A differential level 1 incremental backup contains only blocks modified since the last incremental backup.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Full Backups

A full backup is different from a whole database backup. A full data file backup is a backup that includes every used data block in the file. RMAN copies all blocks into the backup set or image copy, skipping only those data file blocks that have never been used. For a full image copy, the entire file contents are reproduced exactly. A full backup cannot be part of an incremental backup strategy; it cannot be the parent for a subsequent incremental backup.

Incremental Backups

An incremental backup is either a level 0 backup, which includes every block in the data files except blocks that have never been used, or a level 1 backup, which includes only those blocks that have been changed since a previous backup was taken. A level 0 incremental backup is physically identical to a full backup. The only difference is that the level 0 backup can be used as the base for a level 1 backup, but a full backup can never be used as the base for a level 1 backup.

Incremental backups are specified using the `INCREMENTAL` keyword of the `BACKUP` command. You specify `INCREMENTAL LEVEL [0 | 1]`.

RMAN can create multilevel incremental backups as follows:

- **Differential:** Is the default type of incremental backup that backs up all blocks changed after the most recent incremental backup at either level 1 or level 0
- **Cumulative:** Backs up all blocks changed after the most recent backup at level 0

Examples

- To perform an incremental backup at level 0, use the following command:
RMAN> BACKUP INCREMENTAL LEVEL 0 DATABASE;
- To perform a differential incremental backup, use the following command:
RMAN> BACKUP INCREMENTAL LEVEL 1 DATABASE;
- To perform a cumulative incremental backup, use the following command:
RMAN> BACKUP INCREMENTAL LEVEL 1 CUMULATIVE DATABASE;

RMAN makes full backups by default if neither FULL nor INCREMENTAL is specified. Unused block compression causes never-written blocks to be skipped when backing up data files to backup sets, even for full backups.

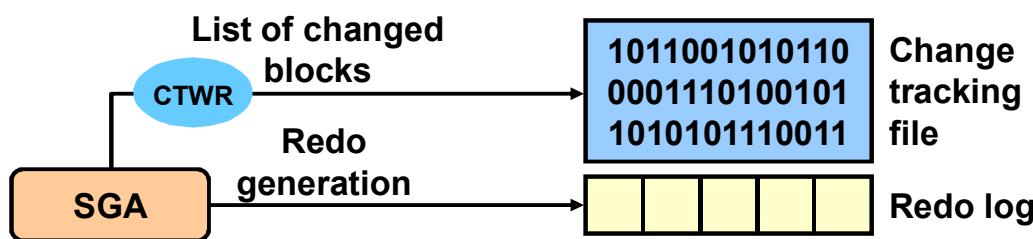
A full backup has no effect on subsequent incremental backups, and is not considered part of any incremental backup strategy, although a full image copy backup can be incrementally updated by applying incremental backups with the RECOVER command. This is covered in the lesson titled “Using RMAN to Perform Recovery.”

Note: It is possible to perform any type of backup (full or incremental) of a database that is in NOARCHIVELOG mode—if, of course, the database is not open. Note also that recovery is limited to the time of the last backup. The database can be recovered to the last committed transaction only when the database is in ARCHIVELOG mode.

Fast Incremental Backup

Implemented by block change tracking, which:

- Maintains a record of what blocks have changed since the last backup
- Writes this record to a file, as redo is generated
- Is automatically accessed when a backup is done, making the backup run faster



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The goal of an incremental backup is to back up only those data blocks that have changed since a previous backup. You can use RMAN to create incremental backups of data files, tablespaces, or the whole database. When making an incremental backup, RMAN scans each block of the data files to see which has changed since the last backup. That makes the backup smaller because only changed blocks are backed up. It also makes recovery faster because fewer blocks need to be restored.

You can perform fast incremental backup by enabling block change tracking. Block change tracking writes to a file the physical address of each block that is changed. When it is time to perform the incremental backup, RMAN can look at the block change tracking file, and back up only those blocks referenced there; it does not have to scan every block to see if it has changed since the last backup. This makes the incremental backup faster.

The maintenance of the tracking file is fully automatic and does not require your intervention. The size of the block change tracking file is proportional to the:

- Database size, in bytes
- Number of enabled threads in a RAC environment
- Number of old backups maintained by the block change tracking file

The minimum size for the block change tracking file is 10 MB, and any new space is allocated in 10 MB increments. The Oracle database does not record block change information by default.

Enabling Fast Incremental Backup

The screenshot shows the Oracle Enterprise Manager 10g interface. In the top navigation bar, 'Database Control' is selected. Under 'Backup Settings', the 'Policy' tab is active. In the 'Backup Policy' section, there are several checkboxes and input fields. One checkbox, 'Enable block change tracking for faster incremental backups', has a red box drawn around it, indicating it is the focus of the discussion. Below this checkbox is an input field for 'Block Change Tracking File' with a note: 'Specify a location and file, otherwise an Oracle managed file will be created in the database area.'

```
ALTER DATABASE
{ENABLE|DISABLE} BLOCK CHANGE TRACKING
[USING FILE '...']
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You enable block change tracking from the Database Control home page. Navigate to Availability > Backup Settings > Policy. You do not need to set the block change tracking file destination if the DB_CREATE_FILE_DEST initialization parameter is set because the file is created as an Oracle Managed Files (OMF) file in the DB_CREATE_FILE_DEST location. You can, however, specify the name of the block change tracking file, placing it in any location you choose.

You can also enable or disable this feature by using an ALTER DATABASE command. If the change tracking file is stored in the database area with your database files, then it is deleted when you disable change tracking. You can rename the block change tracking file by using the ALTER DATABASE RENAME command. Your database must be in the MOUNT state to rename the tracking file. The ALTER DATABASE RENAME FILE command updates the control file to refer to the new location. You can use the following syntax to change the location of the block change tracking file:

```
ALTER DATABASE RENAME FILE '...' TO '...';
```

Note: RMAN does not support backup and recovery of the block change tracking file. For this reason, you should not place it in the Flashback Recovery Area.

Monitoring Block Change Tracking

```
SQL> SELECT filename, status, bytes  
2   FROM v$block_change_tracking;
```

```
SQL> SELECT file#, avg(datafile_blocks),  
2           avg(blocks_read),  
3           avg(blocks_read/datafile_blocks)  
4           * 100 AS PCT_READ_FOR_BACKUP,  
5           avg(blocks)  
5   FROM v$backup_datafile  
6  WHERE used_change_tracking = 'YES'  
7  AND incremental_level > 0  
8  GROUP BY file#;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The output of the V\$BLOCK_CHANGE_TRACKING view shows where the block change tracking file is located, the status of block change tracking (ENABLED/DISABLED), and the size (in bytes) of the file.

The query on the V\$BACKUP_DATAFILE view shows how effective the block change tracking is in minimizing the incremental backup I/O (the PCT_READ_FOR_BACKUP column). A high value indicates that RMAN reads most blocks in the data file during an incremental backup. You can reduce this ratio by decreasing the time between the incremental backups.

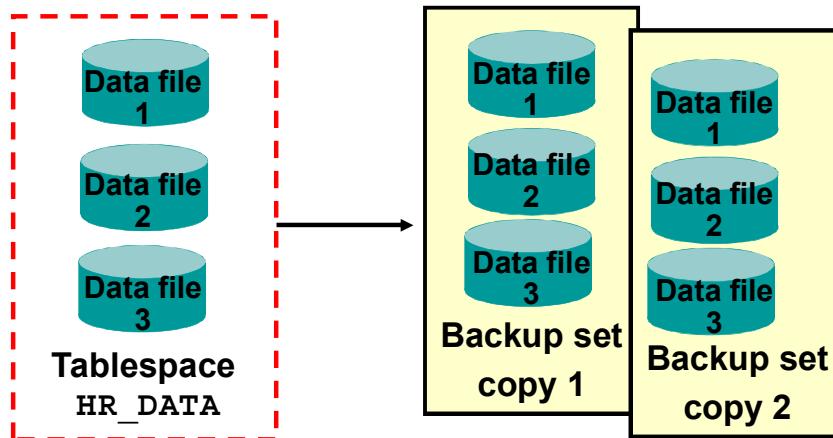
A sample formatted output from the V\$BACKUP_DATAFILE query is shown below:

FILE#	BLOCKS_IN_FILE	BLOCKS_READ	PCT_READ_FOR_BACKUP	BLOCKS_BACKED_UP
1	56320	4480	7	462
2	3840	2688	70	2408
3	49920	16768	33	4457
4	640	64	10	1
5	19200	256	1	91

Creating Duplexed Backup Sets

To create a duplexed backup set, use:

- CONFIGURE . . . BACKUP COPIES
- BACKUP . . . COPIES



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

RMAN can make up to four copies of a backup set simultaneously, each an exact duplicate of the other. A copy of a backup set is a copy of each backup piece in the backup set, with each copy getting a unique copy number (for example, `0tcm8u2s_1_1` and `0tcm8u2s_1_2`).

In most cases, the easiest method of duplexing backup sets is to use `BACKUP . . . COPIES` or `CONFIGURE . . . BACKUP COPIES` to duplex backup sets. For DISK channels, specify multiple values in the `FORMAT` option to direct the multiple copies to different physical disks. For sbt channels, if you use a media manager that supports Version 2 of the SBT API, then the media manager automatically puts each copy onto a separate medium (for example, a separate tape).

Note: The System Backup to Tape (SBT) API is the interface defined for Media Management Library (MML) developers, so that they can provide MMLs that communicate with RMAN.

Note that it is not possible to duplex backup sets to the Flash Recovery Area, and that duplexing applies only to backup sets, not image copies. You receive an error if you specify the `BACKUP . . . COPIES` option when creating image copy backups. The `CONFIGURE . . . BACKUP COPIES` setting is ignored for image copy backups.

Duplexed backup sets are typically used for tape backups.

Creating Duplexed Backup Sets Using CONFIGURE BACKUP COPIES

```
RMAN> CONFIGURE ARCHIVELOG BACKUP COPIES
2> FOR DEVICE TYPE sbt TO 2;
RMAN> CONFIGURE DATAFILE BACKUP COPIES
2> FOR DEVICE TYPE sbt TO 2;
RMAN> BACKUP DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DEVICE TYPE DISK AS COPY DATABASE;
```

Two copies of the backup are made to two different tapes.

Not affected by the COPIES configuration setting. Only one copy is made on disk.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the CONFIGURE . . . BACKUP COPIES command to specify the number of identical backup sets that you want to create on the specified device type. This setting applies to all backups except control file autobackups (because the autobackup of a control file always produces one copy) and backup sets when backed up with the BACKUP BACKUPSET command.

Note: You must have automatic channels configured.

To create a duplexed backup set with CONFIGURE BACKUP COPIES, perform the following steps:

1. Configure the number of copies on the desired device type for data files and archived redo log files.
2. Execute the BACKUP command.
3. Issue a LIST BACKUP command to verify your backup.

Note: The last BACKUP command is not affected by the COPIES configuration setting. It creates a single copy to disk.

Creating Duplexed Backup Sets Using BACKUP COPIES

```
RMAN> BACKUP AS BACKUPSET DEVICE TYPE sbt
2> COPIES 2
3> INCREMENTAL LEVEL 0
4> DATABASE;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

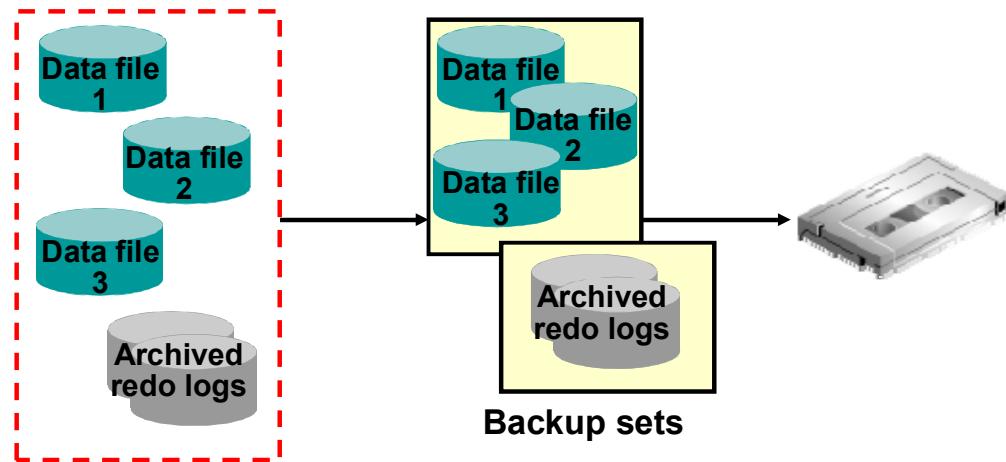
You can use the `BACKUP` command with the `COPIES` option to override other `COPIES` or `DUPLEX` settings to create duplexed backup sets.

To duplex a backup with `BACKUP COPIES`, perform the following steps:

1. Specify the number of identical copies with the `COPIES` option of the `BACKUP` command.
2. Issue a `LIST BACKUP` command to verify your backup.

Creating Backups of Backup Sets

```
RMAN> BACKUP DEVICE TYPE DISK AS BACKUPSET  
2> DATABASE PLUS ARCHIVELOG;  
RMAN> BACKUP DEVICE TYPE sbt BACKUPSET ALL;
```



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the RMAN BACKUP BACKUPSET command to back up previously created backup sets. Only backup sets that were created on device type DISK can be backed up by using RMAN. The backup sets can be backed up to any available device type.

The BACKUP BACKUPSET command uses the default disk channel to copy backup sets from disk to disk. To back up from disk to tape, you must either configure or manually allocate a nondisk channel.

Backing Up Read-Only Tablespaces

Considerations for backing up read-only tablespaces:

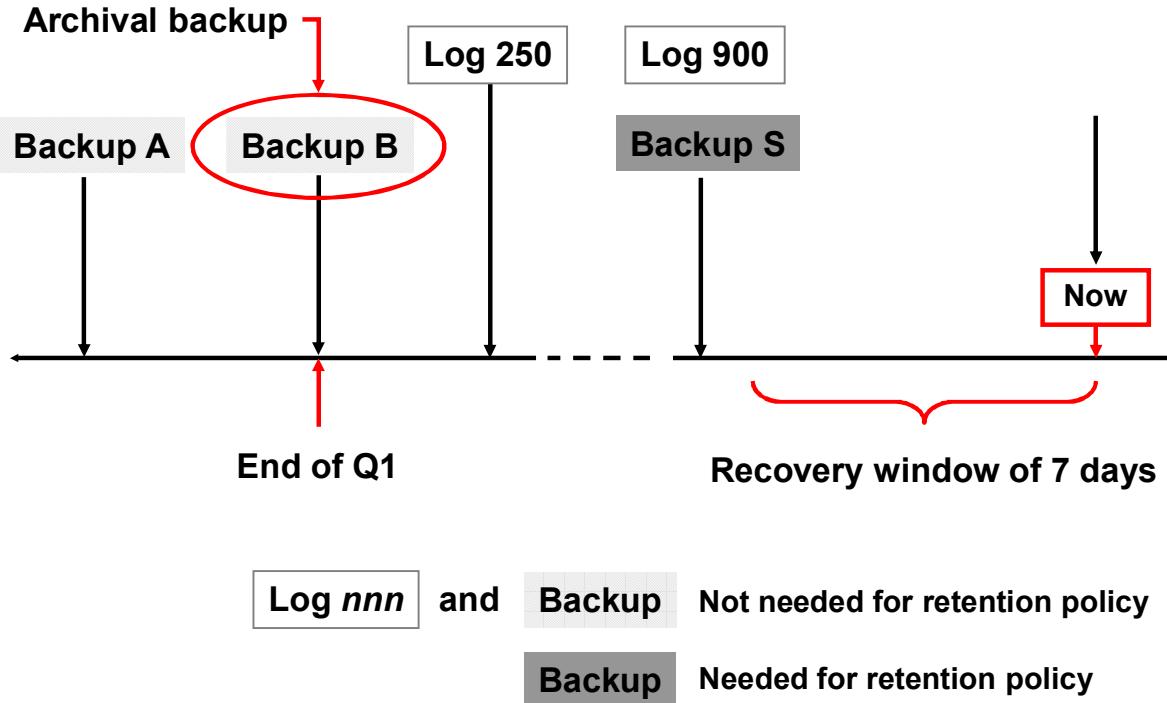
- Backup optimization causes RMAN to back up read-only tablespaces only when there does not exist a backup that satisfies the retention policy.
- If you change the tablespace to read/write, back it up immediately.
- You can use the SKIP READONLY option of the RMAN BACKUP command to skip read-only tablespaces or data files.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Because read-only tablespaces are not being written to, there is no need to continually back them up as you do read/write tablespaces. You can use the SKIP READONLY option of the BACKUP command to let RMAN know to not back up read-only tablespaces.

Archival Backups: Concepts



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

If you need to preserve an online backup for a specified amount of time, RMAN normally assumes you might want to perform point-in-time recovery for any time since that backup to the present. To satisfy this scenario, RMAN keeps the archived logs for that time period. However, you may have a requirement to simply keep the specific backup (and what is necessary to keep it consistent and recoverable) for a specified amount of time—for example, for two years. You do not have the intention of recovering to a point in time since that backup, but you just want to be able to recover to the exact time of the backup, and no later. You also want to maintain a retention policy that keeps your backup area free of clutter, so making it reach back two years is not acceptable. This is a common need, when meeting business or legal requirements for data retention.

An archival backup solves this problem. If you mark a backup as an archival backup, that attribute overrides any configured retention policy for the purpose of this backup. You can retain archival backups such that they are either considered obsolete only after a specific time that you specify, or never considered obsolete. If you want to specify the latter, you need to use a recovery catalog.

The `KEEP` clause creates an archival backup that is a snapshot of the database at a point in time. The only redo logs that are kept are those required to restore this backup to a consistent state. The `RESTORE POINT` clause issued after the backup is completed determines the number of redo logs that are kept (enough to restore the backup to the `RESTORE POINT` time).

An archival backup also guarantees that all of the files needed to restore the backup are included. RMAN includes the data files, archived log files (only those needed to recover an online backup), and the relevant autobackup files. All these files must go to the same media family (or group of tapes).

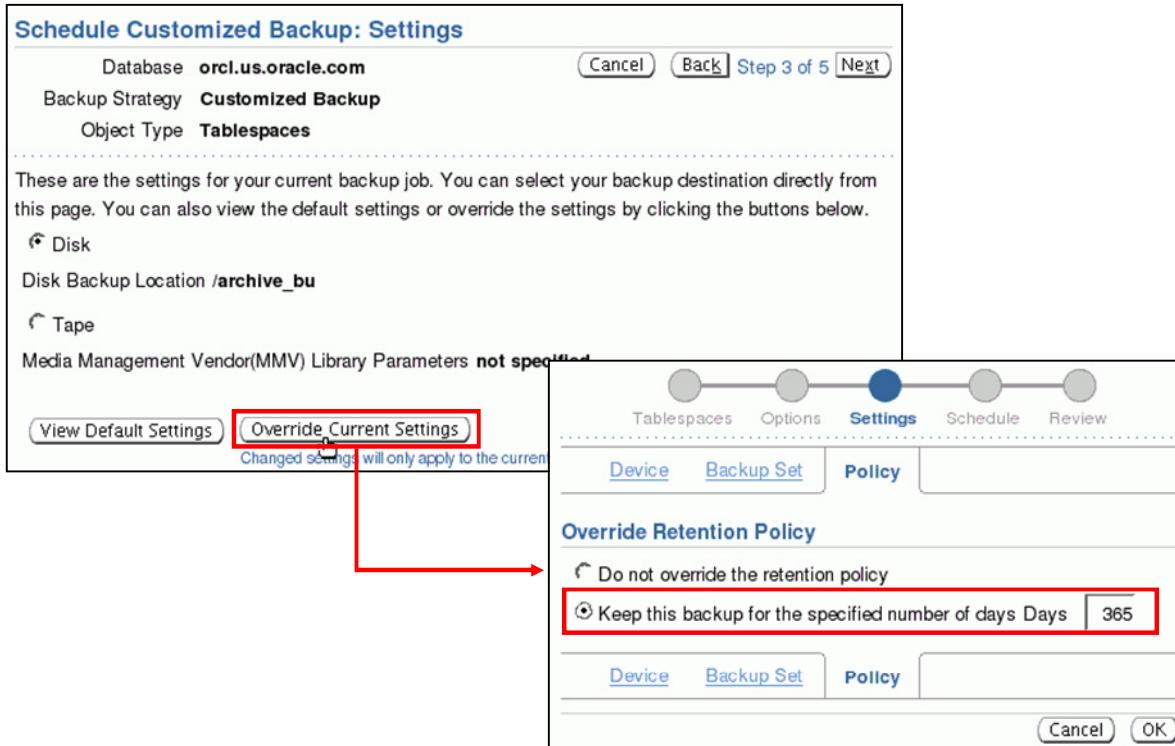
You can also specify a restore point to be created, which has the same SCN as the archival backup. That essentially gives a meaningful name to the point of time the backup was made.

After an archival backup is created, it is retained for as long as specified. Even if you have a much smaller retention window and run the `DELETE OBSOLETE` command, the archival backup remains.

This backup is a snapshot of the database at a point in time, and can be used to restore the database to another host for testing purposes, for example.

Note: Archival backups cannot be written to the Flash Recovery Area. So if you have one, you must provide a `FORMAT` clause to specify a different location.

Creating Archival Backups with EM



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To create an archival backup by using Enterprise Manager, perform the following steps:

1. Select Availability > Schedule Backup > Schedule Customized Backup.
2. Follow the steps of the Schedule Customized Backup Wizard until you are on the Settings page.
3. Click Override Current Settings and then the Policy tab. In the Override Retention Policy section, you can select to keep a backup for a specified number of days. A restore point is generated based on the backup job name. You probably also want to specify a different destination for the backup files; to do this, use the Device tab.

Creating Archival Backups with RMAN

When the database is in the OPEN state, specifying the KEEP clause causes both data file and archive log backup sets to be included.

```
KEEP {FOREVER | UNTIL TIME [=] ' date_string ' }
[RESTORE POINT rsname]
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the following syntax to create an archival backup using RMAN:

```
BACKUP ... KEEP {FOREVER|UNTIL TIME 'SYSDATE + <n>' } RESTORE POINT
<restore_point_name>
```

The UNTIL TIME clause enables you to specify when the archival backup is no longer immune to the retention policy. You can optionally specify FOREVER, which means that the backup is an archival backup until you take some other action to change that.

Optionally, use the RESTORE POINT clause to specify the name of a restore point to be associated with this backup.

Managing Archival Database Backups

1 Archiving a database backup:

```
RMAN> CONNECT TARGET /
RMAN> CONNECT CATALOG rman/rman@catdb
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup'
2> KEEP FOREVER;
```

2 Changing the status of a database copy:

```
RMAN> CHANGE COPY OF DATABASE CONTROLFILE NOKEEP;
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

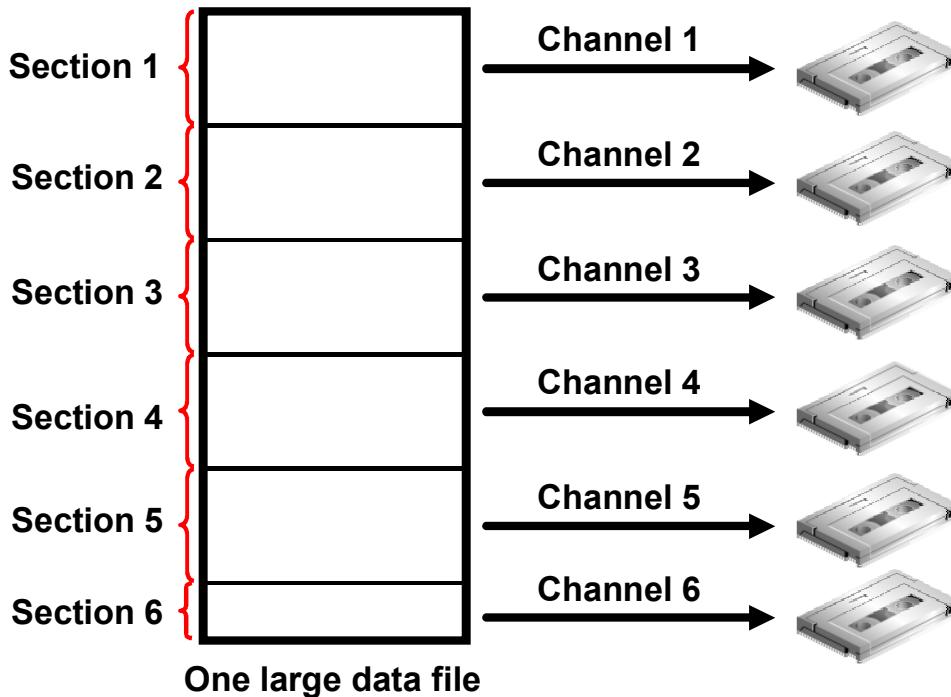
The CHANGE command changes the exemption status of a backup or copy in relation to the configured retention policy. For example, you can specify CHANGE ... NOKEEP, to make a backup that is currently exempt from the retention policy eligible for the OBSOLETE status.

The first example changes a consistent backup into an archival backup, which you plan to store offsite. Because the database is consistent and, therefore, requires no recovery, you do not need to save archived redo logs with the backup.

The second example specifies that any long-term image copies of data files and control files should lose their exempt status and so become eligible to be obsolete according to the existing retention policy. This statement essentially removes the archival attribute from those backup files. If you do not specify a tag, as in this case, then the CHANGE execution applies to all backups of the type specified. You should specify a tag to change only the backup files you intend to change.

Note: The RESTORE POINT option is not valid with the CHANGE command, because there is no way to create the restore point for a time that has already passed (when the backup was created).

Multisection Backups: Overview



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle data files can be up to 128 TB in size. Normally, the smallest unit of an RMAN backup is an entire file. This is not practical with such large files. RMAN can optionally break up large files into sections and back up and restore these sections independently. You do this by creating multisection backups, which break up the files generated for the backup set into separate files. This can be done only with backup sets, not image copies.

Each file section is a contiguous range of blocks of a file. Each file section can be processed independently, either serially or in parallel. Backing up a file into separate sections can improve the performance of the backup operation, and it also allows large file backups to be restarted.

A multisection backup job produces a multipiece backup set. Each piece contains one section of the file. All sections of a multisection backup, except perhaps for the last section, are of the same size. There are a maximum of 256 sections per file.

Note: You should not apply large values of parallelism to back up a large file that resides on a small number of disks, as that would defeat the purpose of the parallel operation; multiple simultaneous accesses to the same disk device would be competing with each other.

This feature is built into RMAN. No installation is required beyond the normal installation of Oracle Database 11g. COMPATIBLE must be set to at least 11.0, because earlier releases cannot restore multisection backups.

Creating RMAN Multisection Backups

```
BACKUP <options> SECTION SIZE <integer> [K | M | G]
```

```
VALIDATE DATAFILE <options> SECTION SIZE <integer> [K | M | G]
```

Example:

```
RMAN> BACKUP DATAFILE 5 SECTION SIZE = 25M TAG 'section25mb';
backing up blocks 1 through 3200
piece handle=/u01/.../o1_mf_nnndf_SECTION25MB_382dryt4_.bkp
tag=SECTION25MB comment=NONE
...
backing up blocks 9601 through 12800
piece handle=/u01/.../o1_mf_nnndf_SECTION25MB_382dsto8_.bkp
tag=SECTION25MB comment=NONE
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The BACKUP and VALIDATE DATAFILE commands accept the following option:

SECTION SIZE <integer> [K | M | G]

Use this to specify your planned size for each backup section. The option is both a backup command- and backup spec-level option, so that you can apply different section sizes to different files in the same backup job.

In the example in the slide, a backup of data file 5 is being taken, and the section size is specified as 25 MB. The data file is 100 MB in size, so four sections are created. Note that, as indicated by the block ranges, block contiguity is maintained as they are written to the section files.

Viewing Metadata About Your Multisection Backup

- The V\$BACKUP_SET and RC_BACKUP_SET views have a MULTI_SECTION column, which indicates whether this is a multisection backup or not.
- The V\$BACKUP_DATAFILE and RC_BACKUP_DATAFILE views have a SECTION_SIZE column, which specifies the number of blocks in each section of a multisection backup. Zero means a whole-file backup.

Compressing Backups

RMAN can perform binary compression on any backup set that is generated.

- It can be performed in addition to unused block compression.
- By default, RMAN uses the BZIP2 compression algorithm.
- BZIP2 generally differs from ZLIB in the following respects:
 - It has a better compression ratio.
 - It is slower.
- No extra steps are required by the database administrator (DBA) to restore a compressed backup.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

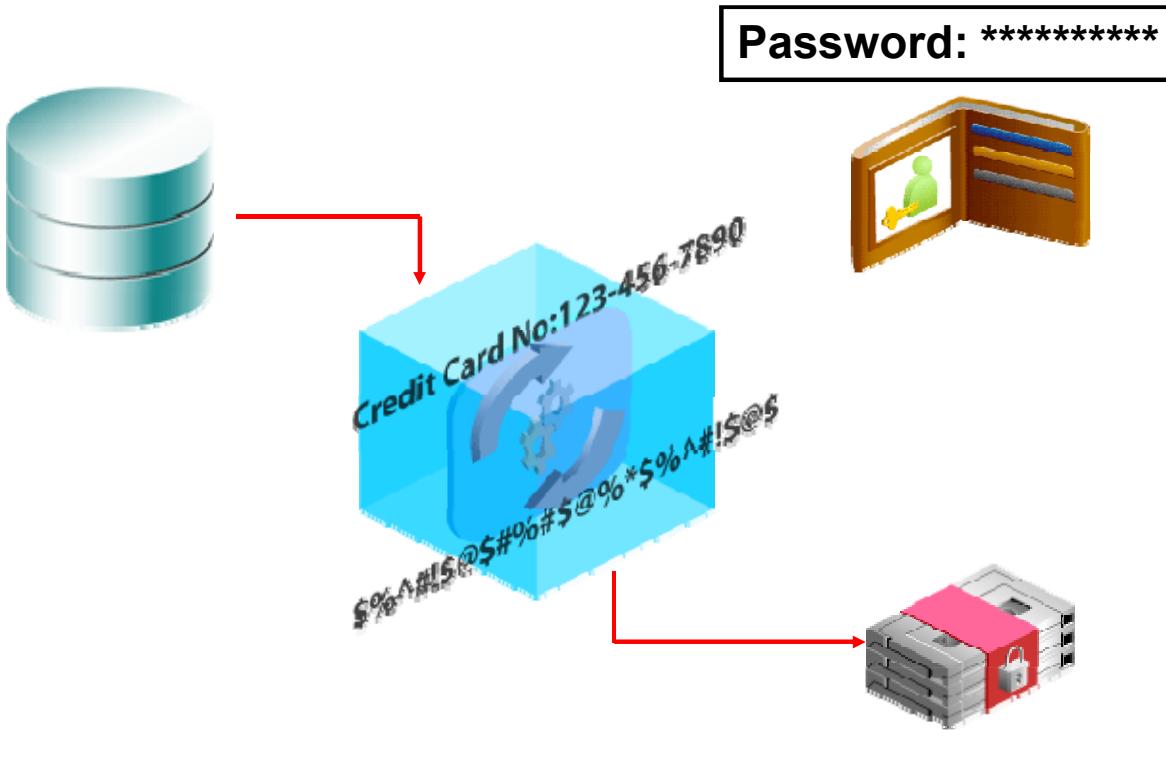
While unused block compression decreases the number of blocks that are written to the backup, binary compression can be used to algorithmically compact the data that is written. The two available compression algorithms are ZLIB and BZIP2.

ZLIB is optimized for CPU efficiency. BZIP2 is optimized for maximum compression. BZIP2 consumes more CPU resource than ZLIB but usually produces more compact backups. The COMPATIBLE initialization parameter must be set to 11.0.0 or higher for ZLIB compression, which requires the Oracle Advanced Compression option.

You do not have to perform any additional steps when restoring a compressed backup. Note, however, that compression and decompression operations require CPU resources. So both creating and restoring a compressed backup will, of course, probably take longer and require more system resources.

When choosing an algorithm, consider your disk space in addition to dynamic system resources such as CPU and memory.

Encrypting Backups



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can encrypt backups in one of three ways:

- **Transparent encryption:** This method uses a wallet, and it is the default mode.
- **Password encryption:** This method of encryption relies on a password. There is no need to configure a wallet. You must know the password that was used for the backup in order to restore.
- **Dual mode encryption:** Both transparent and password encryption are used. In order to restore, either the transparent mode or the password mode can be used. This type of encryption is useful if you usually restore your backups to the local site, but sometimes ships the backups to other sites.

Modify the encryption setting using `SET ENCRYPTION`. Here is an example of password encryption:

```
RMAN> SET ENCRYPTION IDENTIFIED BY mypassword;  
RMAN> BACKUP DATAFILE 5;  
...  
RMAN> SET DECRYPTION IDENTIFIED BY mypassword;
```

RMAN can transparently encrypt data written to backup sets and decrypt those backup sets when they are needed in a RESTORE operation. To create encrypted backups on disk, the database must use the Advanced Security Option. To create encrypted backups directly on tape, RMAN must use the Oracle Secure Backup SBT interface, but does not require the Advanced Security Option.

If you want to use transparent encryption, the wallet must be created and the database configured to use a wallet. This is the same wallet used with Transparent Data Encryption. The wallet must be created before it can be used for either encrypted backups or TDE.

Add an entry to the SQLNET.ORA file:

```
ENCRYPTION_WALLET_LOCATION =
  (SOURCE =
    (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY =
        /oracle/dbsid/admin/pdcs11/wallet)))
```

Create a simple password-protected wallet with the following command:

```
ALTER SYSTEM SET [ENCRYPTION] KEY IDENTIFIED BY "welcome1";
```

Note: Encryption by Oracle Secure Backup (OSB) can be configured so that all backups are encrypted by OSB no matter what type of encryption is requested by the client. For more information about OSB encryption, see the *Secure Backup Administrator's Guide*.

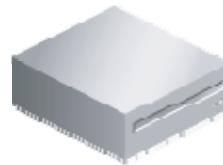
Backing Up Recovery Files

- Back up only the files in the Flash Recovery Area:

```
RMAN> BACKUP RECOVERY AREA
```

- Back up all recovery files:

```
RMAN> BACKUP RECOVERY FILES
```



Flash Recovery Area

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

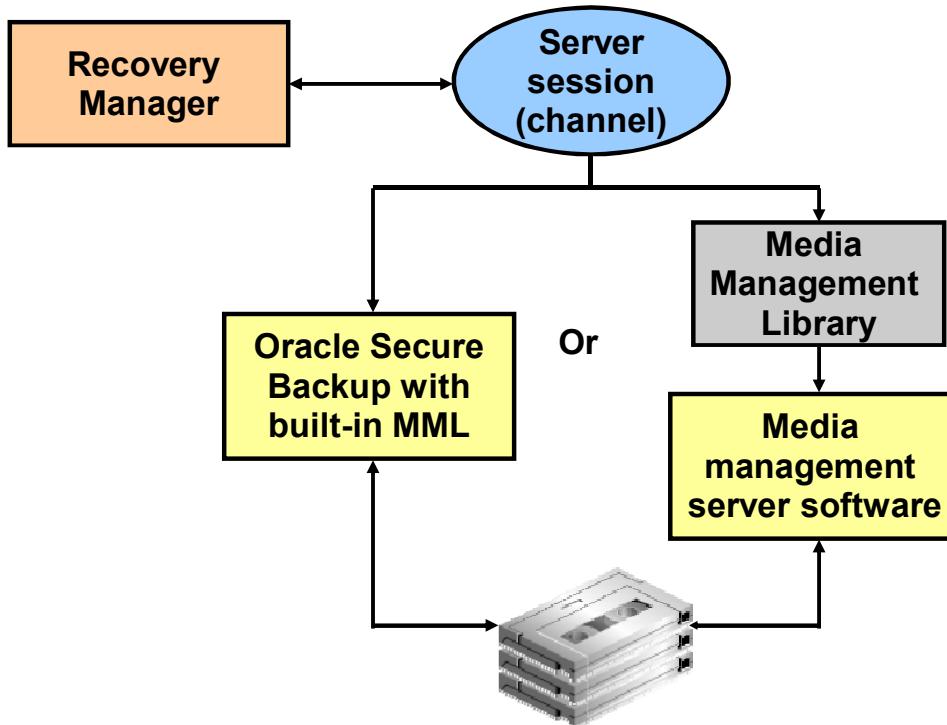
There are two ways to back up recovery data. The `BACKUP RECOVERY AREA` command backs up all files that are found in the current and any previous Flash Recovery Areas. The `BACKUP RECOVERY FILES` command backs up all recovery files, even if they are not in the Flash Recovery Area. You gain added protection from loss by using the latter, which would back up, for example, any copies of control files or data files that are not in the Flash Recovery Area.

By default, backup optimization is in effect for these two commands, even if you have disabled it using the `CONFIGURE` command. This means that the only recovery files that this command backs up are those that are not already backed up. You can force all files to be backed up by using the `FORCE` option.

You cannot specify `DEVICE TYPE DISK` for either of these commands.

Note: RMAN backs up only database files: data files, redo log files, control files, SPFILEs, archive log files, and backups of these files. Placing an operating system file in the Flash Recovery Area causes it to be included with a backup of the recovery area.

Using a Media Manager



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To use tape storage for your database backups, RMAN requires Oracle Secure Backup or a media manager.

A media manager is a utility that loads, labels, and unloads sequential media (such as tape drives) for the purpose of backing up, restoring, and recovering data. The Oracle database server calls Media Management Library (MML) software routines to back up and restore data files to and from media that is controlled by the media manager.

Note that the Oracle database server does not need to connect to the MML software when it backs up to disk.

Oracle Backup Solutions Program (BSP) provides a range of media management products that are compliant with Oracle's MML specification. Software that is compliant with the MML interface enables an Oracle database session to back up data to a media manager and request the media manager to restore backups. Check with your media vendor to determine whether it is a member of Oracle BSP.

Before you can begin using RMAN with a media manager, you must install the media manager software and make sure that RMAN can communicate with it. Instructions for this procedure should be available in the media manager vendor's software documentation.

Depending on the product that you are installing, perform the following basic steps:

1. Install and configure the media management software on the target host or production network.
No RMAN integration is required at this stage.
2. Ensure that you can make non-RMAN backups of operating system files on the target database host. This step makes it easier to troubleshoot problems at a later time. Refer to your media management documentation to learn how to back up files to the media manager.
3. Obtain and install the third-party media management module for integration with the Oracle database. This module must contain the library loaded by the Oracle database server when accessing the media manager.

Backup and Restore Operations Using a Media Manager

The following Recovery Manager script performs a data file backup to a tape drive controlled by a media manager:

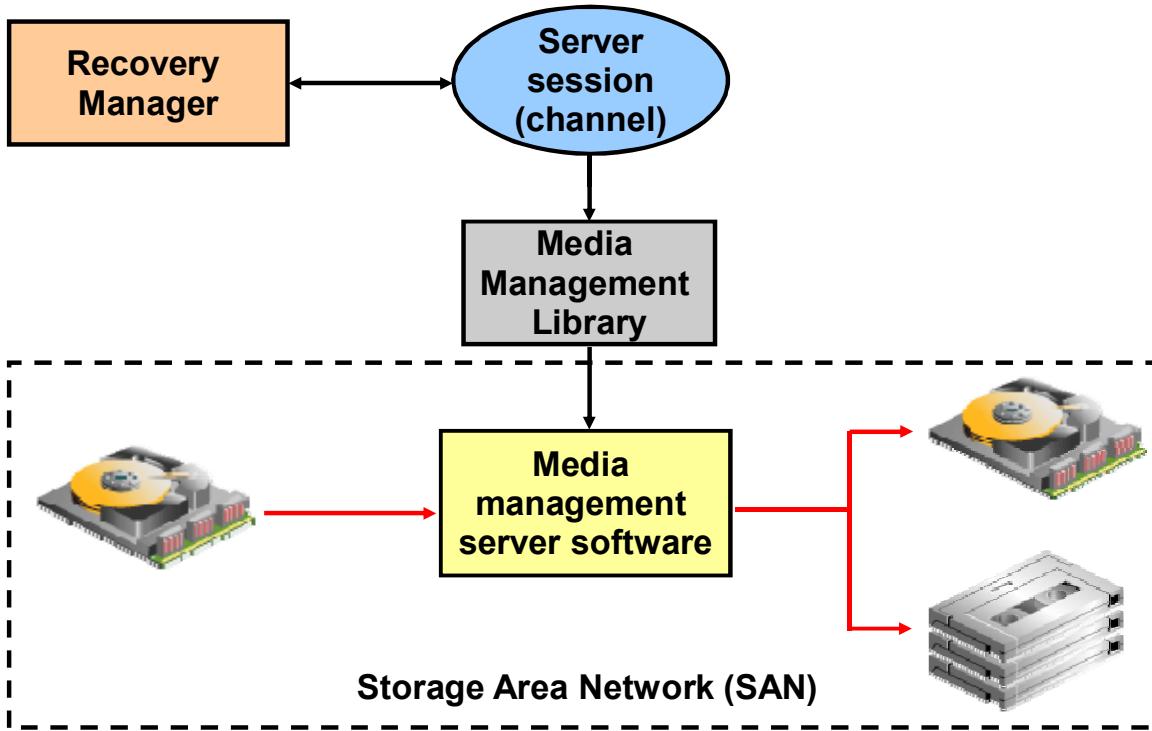
```
run {
  # Allocating a channel of type 'sbt' for serial device
  ALLOCATE CHANNEL ch1 DEVICE TYPE sbt;
  BACKUP DATAFILE 3;
}
```

When Recovery Manager executes this command, it sends the backup request to the Oracle database session performing the backup. The Oracle database session identifies the output channel as a media management device and requests the media manager to load a tape and write the output.

The media manager labels and keeps track of the tape and the names of the files on each tape. The media manager also handles restore operations. When you restore a file, the following steps occur:

1. The Oracle database server requests the restoration of a particular file.
2. The media manager identifies the tape containing the file and reads the tape.
3. The media manager passes the information back to the Oracle database session.
4. The Oracle database server writes the file to disk.

Performing Proxy Copies



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the `PROXY` option of the `RMAN BACKUP` command to request an MML to perform the copy of the files.

Some media management products can completely manage all data movement between Oracle data files and the backup devices. Some products that use high-speed connections between storage and media subsystems can reduce much of the backup load from the primary database server. This is beneficial in that the copying takes place across the SAN instead of the LAN. Also, RMAN is out of the picture at that point, except for communicating status across the LAN to and from the MML.

Creating an Oracle-Suggested Backup

Schedule Backup

Based on your disk and/or tape configuration, Oracle provides an automated backup strategy, or you can develop your own backup strategy with Oracle's automated 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
- ArchiveLogs
- All Recovery Files on Disk

These files include all archive 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:	oracle
* Password:	*****

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Enterprise Manager makes it easy for you to set up an Oracle-suggested backup strategy that protects your data and provides efficient recoverability to any point in the preceding 24 hours, and possibly as far back as 48 hours, depending on when the last backup was created. The Oracle-suggested strategy uses the incremental backup and incrementally updated backup features, providing faster recoverability than is possible when applying database changes from the archived log files.

To establish an Oracle-suggested strategy, navigate to the Maintenance page. In the Backup/Recovery region, select Schedule Backup. The Backup Strategies section enables you to select from the Oracle-suggested backup and Customized backup strategies. The Oracle-suggested strategy takes a full database copy as the first backup. Because it is a whole database backup, you might want to consider taking this at a period of least activity. After that, an incremental backup to disk is taken every day. Optionally, a weekly tape backup can be made, which backs up all recovery-related files.

Because these backups on disk are retained, you can always perform a full database recovery or a point-in-time recovery to any time within the past 24 hours, at the minimum. The recovery time could reach back as far as 48 hours. This is because just before a backup is taken on a given day, the backup from the *beginning* of day $n-1$ still exists.

Managing Backups: Reporting

Use the following RMAN commands to obtain information about your backups:

- **LIST**: Displays information about backup sets, proxy copies, and image copies recorded in the repository
- **REPORT**: Produces a detailed analysis of the repository
- **REPORT NEED BACKUP**: Lists all data files that require a backup
- **REPORT OBSOLETE**: Identifies files that are no longer needed to satisfy backup retention policies



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the RMAN **LIST** command to display information about backup sets, proxy copies, and image copies recorded in the repository. Use this command to list:

- Backups and copies that do not have the **AVAILABLE** status in the RMAN repository
- Backups and copies of data files that are available and can possibly be used in a restore operation
- Backup sets and copies that contain a backup of a specified list of data files or specified tablespaces
- Backup sets and copies that contain a backup of any archived logs with a specified name or range
- Backup sets and copies restricted by tag, completion time, recoverability, or device
- Incarnations of a specified database or of all databases known to the repository
- Stored scripts in the recovery catalog

Use the RMAN **REPORT** command to analyze information in the RMAN repository in more detail.

The **REPORT NEED BACKUP** command is used to identify all data files that need a backup. The report assumes that the most recent backup would be used in the event of a restore.

Using the REPORT OBSOLETE command, you can identify files that are no longer needed to satisfy backup retention policies. By default, the REPORT OBSOLETE command reports which files are obsolete under the currently configured retention policy. You can generate reports of files that are obsolete according to different retention policies by using REDUNDANCY or RECOVERY WINDOW retention policy options with the REPORT OBSOLETE command.

For detailed syntax information, refer to *Oracle Database Backup and Recovery Reference*.

Managing Backups: Dynamic Performance Views

Query the following dynamic performance views in the target database to obtain information about your backups:

- V\$BACKUP_SET: Backup sets created
- V\$BACKUP_PIECE: Backup pieces that exist
- V\$DATAFILE_COPY: Copies of data files on disk
- V\$BACKUP_FILES: Information about all files created when creating backups



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

There are many views that provide backup-related information. The most commonly used ones are shown in the slide.

If you are using a recovery catalog, you can query corresponding views that contain the same information for each target database registered in the recovery catalog database. The corresponding views have the same name, except that the “V\$” is replaced with “RC_”. Also, they are in the schema owned by the recovery catalog owner. For example, the corresponding views in the recovery catalog, showing the information shown in the slide are RC_BACKUP_SET, RC_BACKUP_PIECE, RC_DATAFILE_COPY, and RC_BACKUP_FILES.

To query the RC_BACKUP_FILES view, you must first execute the following in the recovery catalog database:

```
SQL> CALL DBMS_RCMAN.SETDATABASE(null,null,null,<dbid>);
```

where <dbid> is the database ID of a target database.

Using Enterprise Manager to View Backup Reports

Results

Input Summary

Datafile		Control File		SPFile	
Files Backed Up	15	Files Backed Up	3	Files Backed Up	3
Distinct Files	5	Distinct Files	1	Distinct Files	2
Distinct Tablespaces	5	Total Size	28.92M	Total Size	0.01K
Total Size	5.26G	Oldest Checkpoint Time	Aug 14, 2007 8:04:16 PM	Oldest Modification Time	Aug 14, 2007 7:54:20 PM
Oldest Checkpoint Time	Aug 14, 2007 9:19:52 PM	Newest Checkpoint Time	Aug 14, 2007 9:21:59 PM	Newest Modification Time	Aug 14, 2007 9:01:45 PM
Newest Checkpoint Time	Aug 14, 2007 9:21:52 PM				

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 19 (Completed ✓ 15 Failed ✘ 4)

Backup Name	Status	Start Time ▾	Time Taken	Type	Output Devices	Input Size	Output Size	Output Rate (Per Sec)
2007-08-15T00:18:04	COMPLETED	Aug 15, 2007 12:18:10 AM GMT+07:00	00:01:12	DATAFILE FULL	DISK	1.14M	304.00K	4.22K
2007-08-14T19:55:28	COMPLETED	Aug 14, 2007 7:56:20 PM GMT+07:00	01:25:50	DB INCR	DISK	1.86G	1.80G	367.41K
2007-08-14T02:52:20	COMPLETED	Aug 14, 2007 2:52:22 AM GMT+07:00	00:00:09	CONTROLFILE SBT_TAPE		9.63M	0.00K	0.00K

Manage

- [Schedule Backup](#)
- [Manage Current Backups](#)
- [Backup Reports](#) (highlighted)
- [Manage Restore Points](#)
- [Perform Recovery](#)
- [View and Manage Transactions](#)

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use the Backup Report page to display lists of backup jobs that are known to the database through the information recorded about them in the database control file.

You can customize the jobs that appear in the Results table by using the Search fields at the top of the page. The Results table lists basic information about each backup job, such as the Start Time, the Time Taken, and the Status of the backup job. You can also use the Results table to drill down to individual, detailed backup job reports by using the link in the Backup Name column.

You can drill down to a “Summary of Job” page of the backup job by clicking the Status of the job in the Results table, where you can view the contents of the output log.

Click the Backup Name link, and you can use the Backup Report page to display detailed information about that backup. The information displayed on this page is derived from the information recorded in the database control file.

The Backup Report page displays result information in the Result section in various categories, such as Input Summary, containing rollup information about the files that were backed up; Output Summary, containing rollup information about the Backup Sets and Image Copies; and then Inputs and Outputs sections that display tables containing detailed job information about the data files, control files, backup sets, backup pieces, and image copies.

Managing Backups: Cross-Checking and Deleting

Use the following RMAN commands to manage your backups:

- **CROSSCHECK:** Verifies the status of backups and copies recorded in the RMAN repository against media such as disk or tape
- **DELETE EXPIRED:** Removes only files whose status in the repository is EXPIRED
- **DELETE OBSOLETE:** Deletes backups that are no longer needed



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the CROSSCHECK command to ensure that data about backups in the recovery catalog or control file is synchronized with actual files on disk or in the media management catalog. The CROSSCHECK command operates only on files that are recorded in the RMAN repository.

The CROSSCHECK command checks only objects marked AVAILABLE or EXPIRED by examining the files on disk for DISK channels or by querying the media manager for sbt channels. The CROSSCHECK command updates the repository records for any files that it is unable to find to EXPIRED. It does not delete any files that it is unable to find.

The DELETE command can remove any file that the LIST and CROSSCHECK commands can operate on. For example, you can delete backup sets, archived redo logs, and data file copies. The DELETE command removes both the physical file and the catalog record for the file. The DELETE OBSOLETE command deletes backups that are no longer needed. It uses the same REDUNDANCY and RECOVERY WINDOW options as REPORT OBSOLETE.

If you delete backups without using RMAN, you can use the UNCATALOG command to remove the files from the recovery catalog, or you can use the CROSSCHECK and DELETE EXPIRED commands.

For detailed syntax information, refer to *Oracle Database Backup and Recovery Reference*.

Summary

In this lesson, you should have learned how to:

- Create image file backups
- Create a whole database backup
- Create a full database backup
- Enable fast incremental backup
- Create duplex backup sets
- Back up a backup set
- Create an archival backup for long-term retention
- Create a multisection backup
- Create a compressed backup
- Create an encrypted backup
- Report on and maintain backups



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Using RMAN to Perform Recovery

4

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to use RMAN to do the following:

- Perform complete recovery when a critical or noncritical data file is lost
- Recover using incrementally updated backups
- Switch to image copies for fast recovery
- Restore a database onto a new host
- Recover using a backup control file



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Using RMAN RESTORE and RECOVER Commands

- RESTORE command: Restores database files from backup
- RECOVER command: Recovers restored files by applying changes recorded in incremental backups and redo log files

```
RMAN> SQL 'ALTER TABLESPACE inv_tbs OFFLINE IMMEDIATE';
RMAN> RESTORE TABLESPACE inv_tbs;
RMAN> RECOVER TABLESPACE inv_tbs;
RMAN> SQL 'ALTER TABLESPACE inv_tbs ONLINE';
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Reconstructing the contents of an entire database or a part of it from a backup typically involves two phases: retrieving a copy of the data file from a backup, and reapplying changes to the file since the backup from the archived and online redo logs, to bring the database to the desired system change number (SCN) (usually the most recent one).

The RESTORE command retrieves the data file onto disk from a backup location on tape, disk, or other media, and makes it available to the database server. RMAN restores from backup any archived redo logs required during the recovery operation. If backups are stored on a media manager, channels must be configured or allocated for use in accessing backups stored there.

The RECOVER command takes the restored copy of the data file and applies to it the changes recorded in the incremental backups and the redo logs of the database.

For additional information and examples using the RESTORE and RECOVER commands, refer to the *Oracle Database Backup and Recovery User's Guide*.

Performing Recovery Using Enterprise Manager

The Enterprise Manager Recovery Wizard creates and runs an RMAN script to perform the recovery.

The screenshot shows the Oracle Enterprise Manager interface for a database instance named `orcl.us.oracle.com`. The top navigation bar includes links for Home, Performance, Availability (which is highlighted with a red box), Server, Schema, Data Movement, and Software and Support. The Availability tab is active, displaying the 'Backup/Recovery' section. Under 'Backup/Recovery', there are two main categories: 'Setup' (with links for Backup Settings, Recovery Settings, and Recovery Catalog Settings) and 'Manage' (with links for Schedule Backup, Manage Current Backups, Backup Reports, Manage Restore Points, Perform Recovery, and View and Manage Transactions). The 'Perform Recovery' link under 'Manage' is also highlighted with a red box.



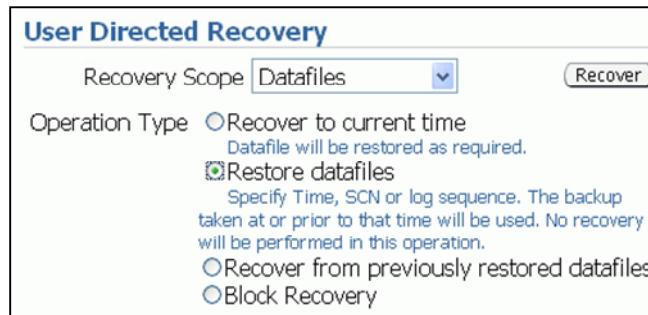
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can also perform complete or incomplete recovery by using the Recovery Wizard available through Enterprise Manager. On the Availability page, click Perform Recovery in the Backup/Recovery section.

Note: An automated method of detecting the need for recovery, and carrying out that recovery makes use of the Data Recovery Advisor, which is covered in the lesson titled “Diagnosing the Database.”

Performing Complete Recovery: Loss of a Noncritical Data File in ARCHIVELOG Mode

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



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

With the database in ARCHIVELOG mode, the loss of any data file not belonging to the SYSTEM or UNDO tablespaces affects only those objects that are in the missing file.

To restore and recover the missing data file using Enterprise Manager, perform the following steps:

1. Click Perform Recovery on the Availability properties page.
2. Select Datafiles as Recovery Scope and “Restore datafiles” as Operation Type.
3. Add all data files that need recovery.
4. Specify from what backup the files are to be restored.
5. Determine whether you want to restore the files to the default location or (if a disk or controller is missing) to a new location.
6. Submit the RMAN job to restore and recover the missing files.

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

Performing Complete Recovery: Loss of a System-Critical Data File in ARCHIVELOG Mode

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

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



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

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).

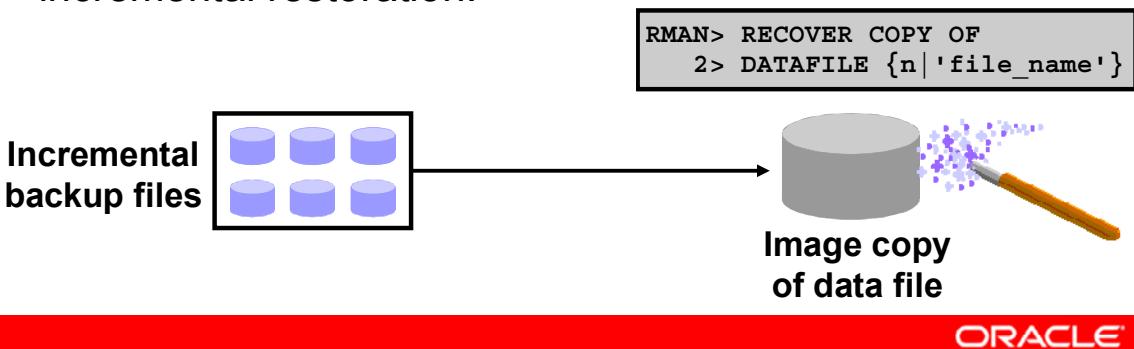
Perform the following steps for complete recovery:

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.

Recovering Image Copies

RMAN can recover image copies by using incremental backups:

- Image copies are updated with all changes up to the incremental backup SCN.
- Incremental backup reduces the time required for media recovery.
- There is no need to perform an image copy after the incremental restoration.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

You can use RMAN to apply incremental backups to data file image copies. With this recovery method, you use RMAN to recover a copy of a data file—that is, you roll forward (recover) the image copy to the specified point in time by applying the incremental backups to the image copy. The image copy is updated with all changes up through the SCN at which the incremental backup was taken. RMAN uses the resulting updated data file in media recovery just as it would use a full image copy taken at that SCN, without the overhead of performing a full image copy of the database every day. The following are the benefits of applying incremental backups to data file image copies:

- You reduce the time required for media recovery (using archive logs) because you need to apply archive logs only since the last incremental backup.
- You do not need to perform a full image copy after the incremental restoration.

If the recovery process fails during the application of the incremental backup file, you simply restart the recovery process. RMAN automatically determines the required incremental backup files to apply, from before the image data file copy until the time at which you want to stop the recovery process. If there is more than one version of an image copy recorded in the RMAN catalog, RMAN automatically uses the latest version of the image copy. RMAN reports an error if it cannot merge an incremental backup file with an image copy.

Recovering Image Copies: Example

If you run these commands daily:

```
RMAN> recover copy of database with tag 'daily_inc';
RMAN> backup incremental level 1 for recover of copy
2> with tag 'daily_inc' database;
```

This is the result:

	RECOVER	BACKUP
Day 1	Nothing	Create image copies
Day 2	Nothing	Create incremental level 1
Day 3 and onward	Recover copies based on incremental	Create incremental level 1



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you run the commands shown in the slide daily, you get continuously updated image copies of all the database data files at any time.

The chart shows what happens for each run. Note that this algorithm requires some priming; the strategy does not come to fruition until after day 3.

Day 1

The RECOVER command does nothing. There exist no image copies to recover yet. The BACKUP command creates the image copies.

Day 2

The RECOVER command, again, does nothing. This is because there is no incremental backup yet. The BACKUP command creates the incremental backup, now that baseline image copies have been created on day 1.

Day 3

The RECOVER command applies the changes from the incremental backup, to the image copies. The BACKUP command takes another incremental backup, which will be used to recover the image copies on day 4. The cycle continues like this.

It is important to use tags when implementing this kind of backup strategy. They serve to link these particular incremental backups to the image copies that are made. Without the tag, the most recent, and possibly the incorrect, incremental backup would be used to recover the image copies.

Performing a Fast Switch to Image Copies

Perform fast recovery by performing the following steps:

1. Take data files offline.
2. Use the SWITCH TO ... COPY command to switch to image copies.
3. Recover data files.
4. Bring data files online.

Now the data files are recovered and usable in their new location.

→ Optionally, do the following to put the files back into their original location:

5. Create an image copy of the data file in the original location.
6. Take data files offline.
7. SWITCH TO ... COPY
8. Recover data files.
9. Bring data files online.

```
SQL> SWITCH DATAFILE 'filename' TO COPY;
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use image copies of data files for fast recovery by performing the following steps:

1. Take the data files offline.
2. Use the SWITCH TO ... COPY command to point to the image copy of the files.
3. Recover the data files.
4. Bring the data files online.

At this point, the database is usable, and the data files are recovered. But, if you want to put the data files back into their original location, proceed with the following steps:

5. Create an image copy of the data files in the original location using the BACKUP AS COPY command.
6. Take the data files offline.
7. Switch to the copy you made in step 5 using the SWITCH TO COPY command.
8. Recover the data files.
9. Bring the data files online.

You can recover data files, tablespaces, tempfiles, or the entire database with this command. The files being switched to must be image copies.

Using SET NEWNAME for Switching Files

For RUN blocks, you can use the SET NEWNAME command to prepare for SWITCH commands.

```
RUN
{
  ALLOCATE CHANNEL dev1 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev2 DEVICE TYPE sbt;
  SQL "ALTER TABLESPACE users OFFLINE IMMEDIATE";
  SET NEWNAME FOR DATAFILE
    '/disk1/oradata/prod/users01.dbf'
    TO '/disk2/users01.dbf';
  RESTORE TABLESPACE users;
  SWITCH DATAFILE ALL;
  RECOVER TABLESPACE users;
  SQL "ALTER TABLESPACE users ONLINE";
}
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The SET NEWNAME command can be used only inside a RUN block. It prepares a name mapping for subsequent operations. In the example in the slide, the SET NEWNAME command defines the location where a restore operation of that data file will be written. When the RESTORE command executes, the users01.dbf data file is restored to /disk2/users01.dbf. It is written there, but the control file is still not pointing to that location. The SWITCH command causes the control file to be updated with the new location.

Performing Restore and Recovery of a Database in NOARCHIVELOG Mode

- If the database is in NOARCHIVELOG mode, and any data file is lost, perform the following tasks:
 - Shut down the instance if it is not already down.
 - Restore the entire database, including all data and control files, from the backup.
 - Open the database.
- Users must reenter all changes made since the last backup.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

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. If the lost data file belongs to a read-only tablespace, you need to restore only that file.

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.

For this type of recovery, use the RESTORE and RECOVER commands, or perform the following tasks in Enterprise Manager:

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.

Creating Restore Points

A restore point provides a name to a point in time:

- Now:

```
SQL> CREATE RESTORE POINT before_mods;
```

- Some time in the past:

```
SQL> CREATE RESTORE POINT end_q1 AS OF SCN 100;
```

Timeline



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can give a name to a particular point in time, or an SCN number. This is useful for future reference, when performing point-in-time recovery or flashback operations.

The first example in the slide creates a restore point that represents the present point in time. If you were about to apply an update of an application or data in the database, and you wanted to refer back to this state of the database, you could use the BEFORE_MODS restore point.

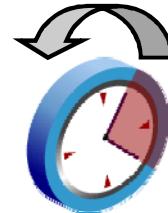
The second example in the slide creates a restore point representing a past SCN, 100. This restore point can be used the same way as the previous one.

Normally, restore points are maintained in the database for at least as long as specified by the CONTROL_FILE_RECORD_KEEP_TIME initialization parameter. However, you can use the PRESERVE option when creating a restore point, which causes the restore point to be saved until you explicitly delete it.

Performing Incomplete Recovery

Perform server-managed incomplete recovery by doing the following:

1. Determine the target point of the restore: SCN, time, restore point, or log sequence number.
2. Set the NLS environment variables appropriately.
3. Mount the database.
4. Prepare and run a `RUN` block, using the `SET UNTIL`, `RESTORE`, and `RECOVER` commands.
5. Open the database in `READONLY` mode, and verify that the recovery point is what you wanted.
6. Open the database using `RESETLOGS`.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can perform server-managed incomplete recovery using the following steps. The database must be in `ARCHIVELOG` mode.

1. Determine the restore target. This can be in terms of a date and time, an SCN, restore point, or log sequence number. For example, if you know that some bad transactions were submitted at 3:00 PM yesterday, then you can choose 2:59 PM yesterday as the target restore point time.
2. Set the National Language Support (NLS) OS environment variables, so that the time constants you provide to RMAN are formatted correctly. These are some example settings:

```
$ export NLS_LANG = american_america.us7ascii
$ export NLS_DATE_FORMAT = "yyyy-mm-dd:hh24:mi:ss"
```
3. Mount the database. If it is open, you have to shut it down first, as in this example:

```
RMAN> shutdown immediate
RMAN> startup mount
```

4. Create a RUN block and run it. The RECOVER and RESTORE commands should be in the same RUN block so that the UNTIL setting applies to both. For example, if you choose to recover to a particular SCN, the RESTORE command needs to know that value so it restores files from backups that are sufficiently old—that is, backups that are from before that SCN. Here is an example of a RUN block:

```
RUN
{
    SET UNTIL TIME '2007-08-14:21:59:00';
    RESTORE DATABASE;
    RECOVER DATABASE;
}
```

5. As soon as you open the database for read/write, you have committed to the restore you just performed. So, first, open the database READ ONLY, and view some data, to check whether the recovery did what you expected:

```
RMAN> SQL 'ALTER DATABASE OPEN READ ONLY';
```

6. If satisfied with the results of the recovery, open the database with the RESETLOGS option, as shown:

```
RMAN> ALTER DATABASE OPEN RESETLOGS;
```

Performing Recovery with a Backup Control File

- Restore and mount a backup control file when all copies of the current control file are lost or damaged.
- Execute the RECOVER command after restoring the backup control file.
- Open the database with the RESETLOGS option after performing complete or point-in-time recovery.

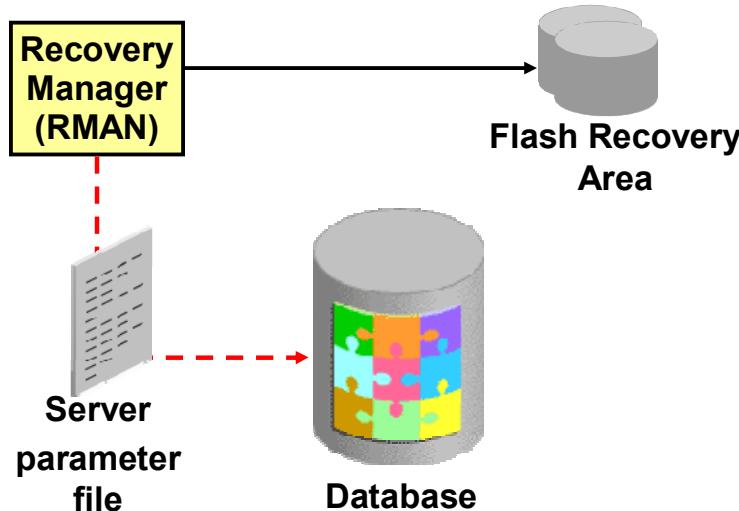


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you have lost all copies of the current control file, you must restore and mount a backup control file before performing recovery. Your recovery operation may be to recover lost data files or it may be to simply recover the control file. If you are using a recovery catalog, the process is identical to recovery with a current control file because RMAN can use the recovery catalog to obtain RMAN metadata.

Restoring the Server Parameter File from the Control File Autobackup

```
RMAN> STARTUP FORCE NOMOUNT;
RMAN> RESTORE SPFILE FROM AUTOBACKUP;
RMAN> STARTUP FORCE;
```



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you have lost the server parameter file, you can restore it from the autobackup. The procedure is similar to restoring the control file from autobackup. If the autobackup is not in the Flash Recovery Area, set the database identifier (DBID) for your database. Issue the RESTORE SPFILE FROM AUTOBACKUP command.

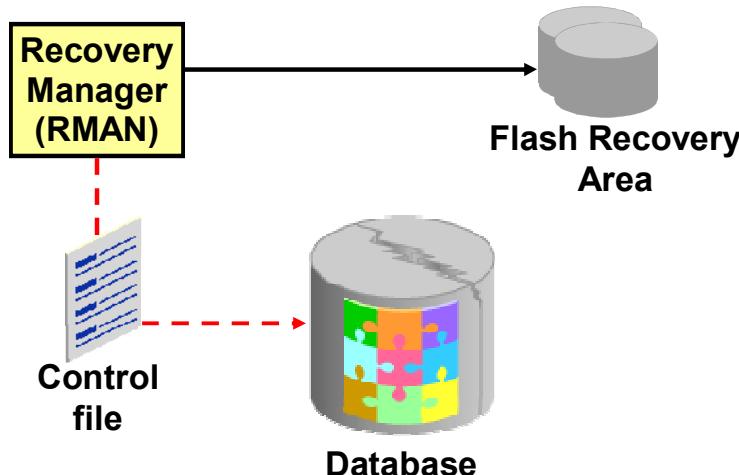
If you are restoring the SPFILE to a nondefault location, specify the command as follows:

```
RESTORE SPFILE TO <file_name> FROM AUTOBACKUP
```

If you are restoring the server parameter file from the Flash Recovery Area, specify the command as follows:

```
RMAN> run {
2> restore spfile from autobackup
3> recovery area = '<flash recovery area destination>'
4> db_name = '<db_name>';
5> }
```

Restoring the Control File from Autobackup



```
RMAN> STARTUP NOMOUNT;
RMAN> RESTORE CONTROLFILE FROM AUTOBACKUP;
RMAN> ALTER DATABASE MOUNT;
RMAN> RECOVER DATABASE;
RMAN> ALTER DATABASE OPEN RESETLOGS;
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you are not using a recovery catalog, you should have autobackup of the control file configured, so that you are able to quickly restore the control file if needed. The commands used for restoring your control file are the same, whether or not you are using a Flash Recovery Area. However, if you are using a Flash Recovery Area, RMAN implicitly cross-checks backups and image copies listed in the control file, and catalogs any files in the Flash Recovery Area not recorded in the restored control file, improving the usefulness of the restored control file in the restoration of the rest of your database.

Use the commands shown in the slide to recover from lost control files. First, start the instance in NOMOUNT mode. It cannot be mounted because there is no control file. Restore the control file from backup. Now that there is a control file, you can mount the database. You must now recover the database, because you now have a backup control file that contains information about an older version of the database. After recovering the database, you can open it. You must specify RESETLOGS because the new control file represents a different instantiation of the database.

Note: Tape backups are not automatically cross-checked after the restoration of a control file. If you are using tape backups, then after restoring the control file and mounting the database, you must cross-check the backups on tape.

To restore the control file from an autobackup, the database must be in a NOMOUNT state. If the autobackup is not in the Flash Recovery Area, you must set the DBID before issuing the RESTORE CONTROLFILE FROM AUTOBACKUP command, as shown in the following example:

```
RMAN> SHUTDOWN ABORT;  
RMAN> STARTUP NOMOUNT;  
RMAN> SET DBID 1090770270;  
RMAN> RESTORE CONTROLFILE FROM AUTOBACKUP;
```

RMAN searches for a control file autobackup. If one is found, RMAN restores the control file from that backup to all the control file locations listed in the CONTROL_FILES initialization parameter.

If you have a recovery catalog, you do not have to set the DBID or use the control file autobackup to restore the control file. You can use the RESTORE CONTROLFILE command with no arguments:

```
RMAN> RESTORE CONTROLFILE;
```

The instance must be in the NOMOUNT state when you perform this operation, and RMAN must be connected to the recovery catalog. The restored control file is written to all locations listed in the CONTROL_FILES initialization parameter.

Use the RESTORE CONTROLFILE... TO <destination> command to restore the control file to a nondefault location.

If you have also lost the SPFILE for the database and need to restore it from the autobackup, the procedure is similar to restoring the control file from autobackup. You must first set the DBID for your database, and then use the RESTORE SPFILE FROM AUTOBACKUP command.

After you have started the instance with the restored server parameter file, RMAN can restore the control file from the autobackup. After you restore and mount the control file, you have the backup information necessary to restore and recover the database.

After restoring the control files of your database from backup, you must perform complete media recovery and then open your database with the RESETLOGS option.

Using Incremental Backups to Recover a Database in NOARCHIVELOG Mode

Use incremental backups to perform limited recovery of a database in NOARCHIVELOG mode.

```
STARTUP FORCE NOMOUNT;
RESTORE CONTROLFILE;
ALTER DATABASE MOUNT;
RESTORE DATABASE;
RECOVER DATABASE NOREDO;
ALTER DATABASE OPEN RESETLOGS;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can perform limited recovery of a NOARCHIVELOG mode database by using incremental backups. The incremental backups must be consistent backups.

If you have taken incremental backups, RMAN will use your level 0 and level 1 backups to restore and recover the database.

You must specify the NOREDO option on the RECOVER DATABASE command if the online redo log files are lost or if the redo cannot be applied to the incremental backups. If you do not specify the NOREDO option, RMAN searches for the online redo log files after applying the incremental backups. If the online redo log files are not available, RMAN issues an error message.

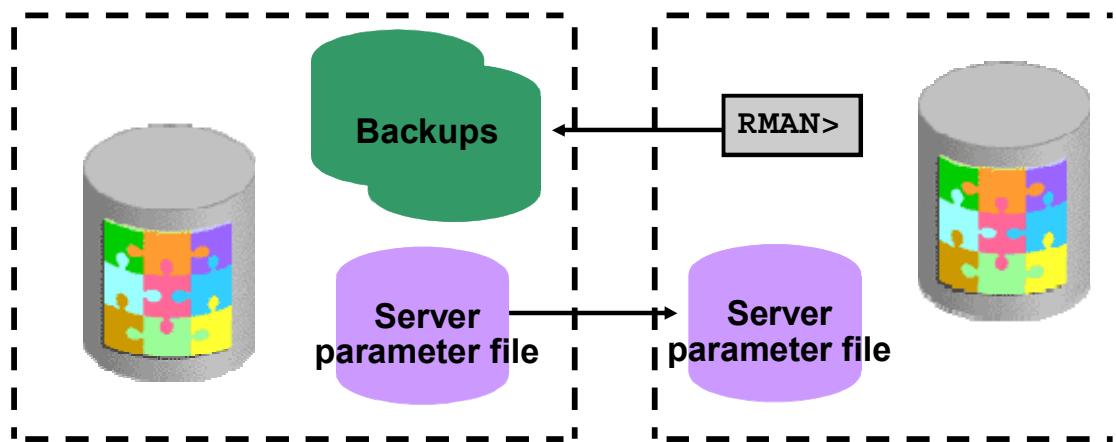
If the current online redo log files contain all changes since the last incremental backup, you can issue the RECOVER DATABASE command without the NOREDO option and the changes will be applied.

Note: You need to restore the control file only if it is not current.

Restoring and Recovering the Database on a New Host

Use the procedure to:

- Perform test restores
- Move a production database to a new host



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the procedure described on the following pages to perform test restores. You can also use it to move a production database to a new host.

The DBID for the restored test database is the same as the DBID of the original database. If you are using a recovery catalog and connect to the test database and the recovery catalog database, the recovery catalog is updated with information about the test database. This can impact RMAN's ability to restore and recover the source database.

You should create a duplicate database using the RMAN DUPLICATE command if your goal is to create a new copy of your target database for ongoing use on a new host. The duplicate database is assigned a new DBID that allows it to be registered in the same recovery catalog as the original target database.

Preparing to Restore the Database to a New Host

To prepare to restore a database, perform the following steps:

1. Record the database identifier (DBID) of your source database.
2. Copy the source database initialization parameter file to the new host.
3. Ensure that source backups, including the control file autobackup, are accessible on the restore host.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Perform the steps listed in the slide to prepare for the restore of the database to a new host.

Note: If you are performing a test restore, do not connect to the recovery catalog when restoring the data files. If you connect to the recovery catalog, RMAN records information about the restored data files in the recovery catalog and considers the restored database as the current target database. If your control file is not large enough to contain all of the RMAN repository data on the backups you need to restore and you must use a recovery catalog, then export the catalog and import it into a different schema or database. Use the copied recovery catalog for the test restore.

Restoring the Database to a New Host

Perform the following steps on the restore host to restore the database:

1. Configure the ORACLE_SID environment variable.
2. Start RMAN and connect to the target instance in NOCATALOG mode.
3. Set the database identifier (DBID).
4. Start the instance in NOMOUNT mode.
5. Restore the server parameter file from the backup sets.
6. Shut down the instance.
7. Edit the restored initialization parameter file.
8. Start the instance in NOMOUNT mode.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Perform the following steps on the restore host to restore the database:

1. Configure the ORACLE_SID environment variable as shown in the following example:
 \$ setenv ORACLE_SID orcl
2. Start RMAN and connect to the target instance. Do not connect to the recovery catalog as shown in the following example:
 \$ rman TARGET /
3. Set the database identifier (DBID). You can find the DBID of your source database by querying the DBID column in V\$DATABASE:
 RMAN> SET DBID 1090770270;
4. Start the instance in NOMOUNT mode:
 RMAN> STARTUP NOMOUNT

You will receive an error similar to the following because the server parameter file has not been restored. RMAN uses a “dummy” parameter file to start the instance.

```
startup failed: ORA-01078: failure in processing system  
parameters
```

5. Restore the server parameter file from the backup sets and shut down the instance as shown in the example:

```
RESTORE SPFILE TO PFILE '?/oradata/test/initorcl.ora' FROM  
AUTOBACKUP;
```

6. Shut down the instance:

```
SHUTDOWN IMMEDIATE;
```

7. Edit the restored initialization parameter file to change any location-specific parameters, such as those ending in _DEST, to reflect the new directory structure.

8. Start the instance in NOMOUNT mode by using your edited text initialization parameter file:

```
RMAN> STARTUP NOMOUNT  
> PFILE='?/oradata/test/initorcl.ora';
```

Restoring the Database to a New Host

9. Create a RUN block to:
 - Restore the control file
 - Mount the database
10. Create the RMAN recovery script to restore and recover the database.
11. Execute the RMAN script.
12. Open the database with the RESETLOGS option.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

9. Create a RUN block to restore the control file from an autobackup and mount the database as shown in the example:

```
RUN
{
  RESTORE CONTROLFILE FROM AUTOBACKUP;
  ALTER DATABASE MOUNT;
}
```
10. Query V\$DATAFILE on your new host to determine the database file names as recorded in the control file. Create the RMAN recovery script to restore and recover the database, performing the following steps as appropriate:
 - a. Use the SET NEWNAME command to specify the path on your new host for each of the data files that is restored to a different destination than on the original host.
 - b. Use the SQL ALTER DATABASE RENAME FILE command to specify the path for the online redo log files.
 - c. Include the SET UNTIL command to limit recovery to the end of the archived redo log files.
 - d. Include the SWITCH command so that the control file recognizes the new path names as the correct names for the data files.

An example of a recovery script follows:

```
RUN
{
SET NEWNAME FOR DATAFILE 1 TO '?/oradata/test/system01.dbf';
SET NEWNAME FOR DATAFILE 2 TO '?/oradata/test/undotbs01.dbf';
SET NEWNAME FOR DATAFILE 3 TO '?/oradata/test/sysaux.dbf';
SET NEWNAME FOR DATAFILE 4 TO '?/oradata/test/users01.dbf';
SET NEWNAME FOR DATAFILE 5 TO '?/oradata/test/example01.dbf';
SQL "ALTER DATABASE RENAME FILE
'#/u01/app/oracle/oradata/orcl/redo01.log'
TO '?/oradata/test/redo01.log' ";
SQL "ALTER DATABASE RENAME FILE
'#/u01/app/oracle/oradata/orcl/redo02.log'
TO '?/oradata/test/redo02.log' ";
SQL "ALTER DATABASE RENAME FILE
'#/u01/app/oracle/oradata/orcl/redo03.log'
TO '?/oradata/test/redo03.log' ";
SET UNTIL SCN 4545727;
RESTORE DATABASE;
SWITCH DATAFILE ALL;
RECOVER DATABASE;
}
```

11. Execute the recovery script.

12. Open the database with the RESETLOGS option:

```
RMAN> ALTER DATABASE OPEN RESETLOGS;
```

After you have completed your test, you can shut down the test database instance and delete the test database with all its files.

Performing Disaster Recovery

- Disaster implies the loss of the entire target database, the recovery catalog database, all current control files, all online redo log files, and all parameter files.
- Disaster recovery includes the restoration and recovery of the target database.
- Minimum required set of backups:
 - Backups of data files
 - Corresponding archived redo logs files
 - At least one control file autobackup



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Disaster recovery includes the restoration and recovery of the target database after the loss of the entire target database, all current control files, all online redo log files, all parameter files, and the recovery catalog database (if applicable).

To perform disaster recovery, the following backups are required as a minimum:

- Backups of data files
- Corresponding archived redo logs generated after the time of the backup
- At least one autobackup of the control file

Note: For information about how Oracle Data Guard can provide complete disaster protection, refer to *Oracle Data Guard Concepts and Administration*.

Performing Disaster Recovery

Basic procedure:

- Restore an autobackup of the server parameter file.
- Start the target database instance.
- Restore the control file from autobackup.
- Mount the database.
- Restore the data files.
- Recover the data files.
- Open the database with the RESETLOGS option.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The basic procedure for performing disaster recovery is outlined in the slide. After you have mounted the database, follow the steps for performing recovery with a backup control file.

Summary

In this lesson, you should have learned how to use RMAN to do the following:

- Perform complete recovery when a critical or noncritical data file is lost
- Recover using incrementally updated backups
- Switch to image copies for fast recovery
- Restore a database onto a new host
- Recover using a backup control file



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

SQL Performance Analyzer

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

- Identify the benefits of using SQL Performance Analyzer
- Describe the SQL Performance Analyzer workflow phases
- Use SQL Performance Analyzer to ascertain performance gains following a database change



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Challenges Faced by DBAs When Performing Changes

- Maintaining service-level agreements through changes to hardware or software configurations
- Offering production-level workload environment for testing purposes
- Effectively forecasting and analyzing impact on SQL performance



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Large business-critical applications are complex and have highly varying load and usage patterns. At the same time, these business systems are expected to provide certain service-level guarantees in terms of response time, throughput, uptime, and availability. Any change to a system (such as upgrading the database or modifying the configuration) often necessitates extensive testing and validation before these changes can make it to the production system. To be confident before moving to a production system, the database administrator (DBA) must expose a test system to a workload very similar to the workload to be experienced in a production environment. It is also beneficial for the DBA to have an effective way to analyze the impact of system-level changes on the overall SQL performance so that any required tuning changes can be performed before production.

Change Is the Only Constant

- Change is the most common cause of instability.



- Enterprise production systems are complex.
- Actual workloads are difficult to simulate.



possible!

- Realistic testing before production is impossible.



- Reluctance to make changes
- Inability to adopt new competitive technologies

Preserve order amid change.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Database 11g is designed for data center environments that are rapidly evolving and changing to keep up with business demands, enabling DBAs to manage change effectively and efficiently. Building on the self-managing capabilities of Oracle Database 10g, Oracle Database 11g offers significant advances in the areas of automatic diagnostics, supportability, and change management.

Oracle DBAs and information technology managers are leading the key initiatives in data centers today. Some of these data center initiatives are moving to low-cost computing platforms (such as Oracle Enterprise Linux) and simplifying storage management by using ASM. DBAs need to test the databases by using *realistic workloads* with new operating systems or storage platforms to ensure that the migration is successful.

Today's enterprises must make significant investments in hardware and software to perform the infrastructure changes. For example, if the DBA wants to test the storage management of data files for a database, from file system-based to ASM for a typical J2EE application, the enterprise would need to invest in duplicate hardware for the entire application stack, including the Web server, application server, and database. The organization would also need to invest in expensive testing software to capture the end-user workload.

These purchases make it very expensive for any organization to evaluate and implement changes to their data center infrastructure. Oracle Database 11g addresses this issue with a collection of solutions under the umbrella of "Change Management."

Change Management in Oracle Database 11g

- SQL Performance Analyzer
- SQL Plan Management
- Database Replay



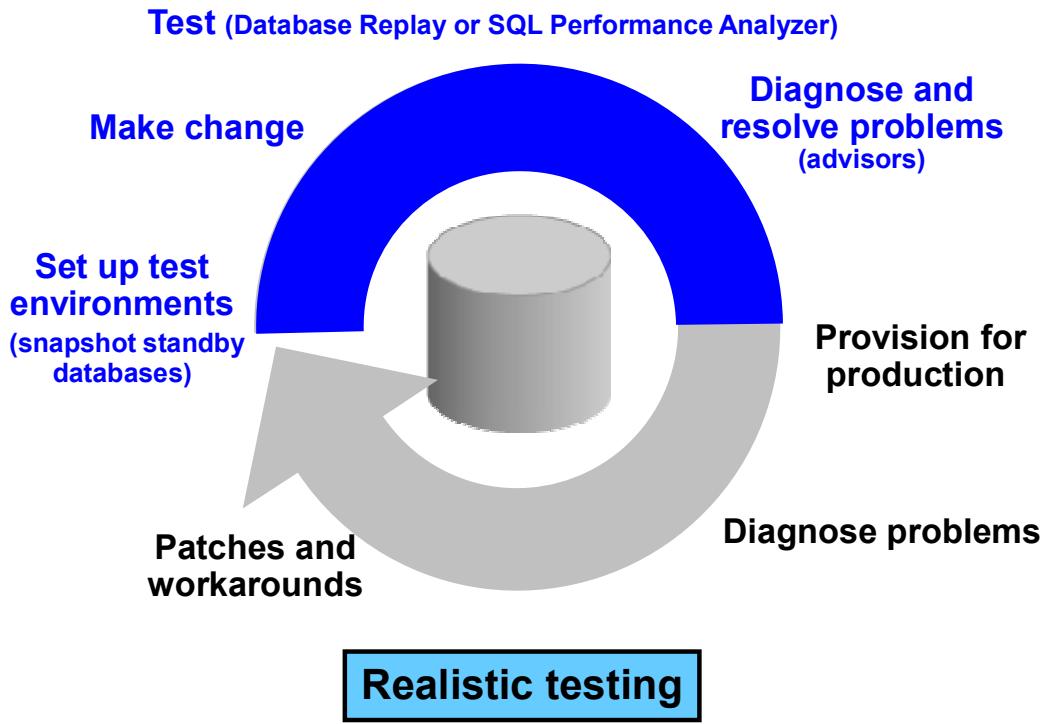
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SQL Performance Analyzer automates the process of assessing the overall effect of a change on the full SQL workload by identifying performance divergence for each SQL statement. Additional information about SQL Performance Analyzer is presented in this lesson.

SQL Plan Management is a new feature introduced with Oracle Database 11g that enables the system to automatically control SQL plan evolution by maintaining SQL plan baselines. SQL Plan Management is discussed in detail in the lesson titled “SQL Plan Management.”

Database Replay enables you to test the impact of a system change by replaying a real-world workload on the test system before it is exposed to a production system.

Life Cycle of Change Management



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

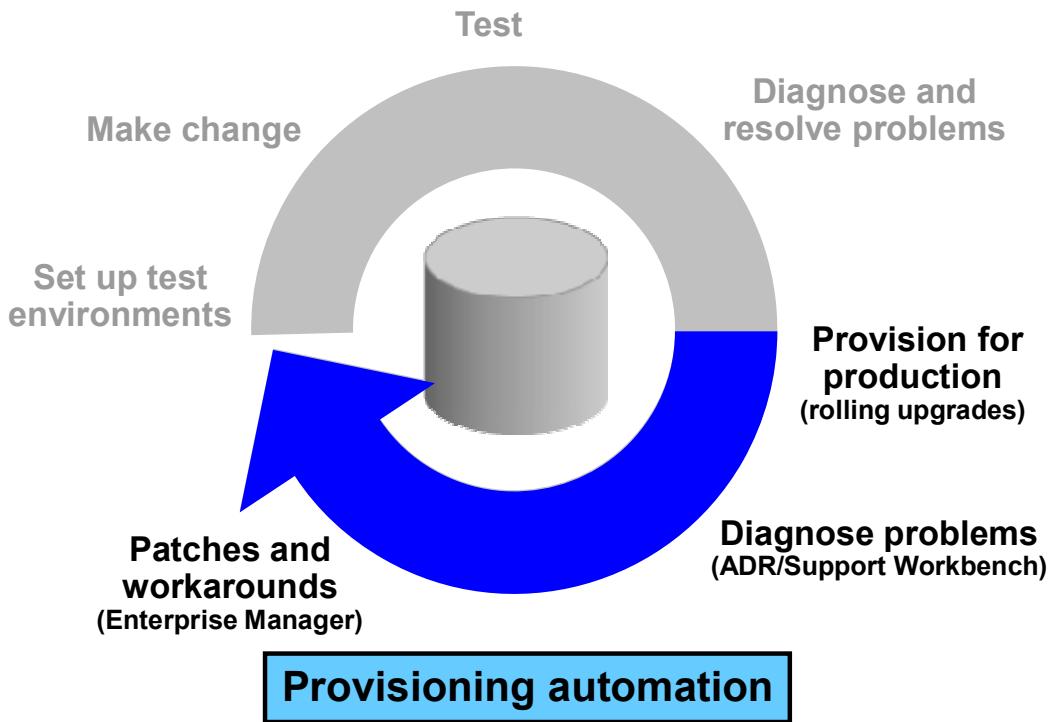
Oracle Database 11g supports realistic testing through the use of snapshot standby databases to set up and test the physical environment. You can open a physical standby database temporarily (that is, activated) for read and write activities such as reporting and testing. Once testing is completed, you can then simply revert to the physical standby mode to allow catch-up to the primary site. This functionality preserves zero data loss and is similar to storage snapshots, but allows for disaster recovery and offers a single copy of storage at the time of testing. Refer to the Oracle Database 11g: New Features for Data Guard Seminar for detailed information on snapshot standby databases.

For enterprises to be able to perform an accurate test of a database environment, it is vital that they be able to reproduce the production scenarios accurately. Database Replay provides further support for realistic testing in Oracle Database 11g. Database Replay is designed to capture client requests on a given database to be reproduced on other copies of production databases. Oracle Enterprise Manager provides an easy-to-use set of steps to set up the capture of a workload.

Some of the changes that a DBA deals with are database upgrades, new tuning recommendations, schema changes, statistics collection, and changes in operating system and hardware. DBAs can use SQL Performance Analyzer to track and forecast SQL performance changes caused by these changes.

If SQL performance has regressed in some of the cases, the DBA can then run the SQL Tuning Advisor to tune the SQL statements.

Life Cycle of Change Management



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When upgrading from Oracle Database 11g, Release 1, you can use the rolling upgrade functionality to ensure that various versions of the software can still communicate with each other. This allows independent nodes of an ASM cluster to be migrated or patched without affecting the availability of the database, thereby providing higher uptime and problem-free migration to new releases. ASM offers further system capacity planning and workload change enhancements (Fast Disk Resync, Preferred Mirror Read). Numerous enhancements to the online functionality (online index reorganization and online table redefinition) further support application change.

Automatic Diagnostic Repository (ADR) is a new system-managed repository for storing and organizing trace files and other error diagnostic data. You get a comprehensive view of all the serious errors encountered by the database, and the relevant data needed for problem diagnosis and eventual resolution. You can also use EM Support Workbench, which provides a simple workflow interface to view and diagnose incident data, and package it for Oracle Support. The Data Recovery Advisor tool can be used to automatically diagnose data failures and report on the appropriate repair option.

Oracle Database 11g Enterprise Manager supports end-to-end automation of patch application on single-instance database homes and rolling patches on clusterware. You no longer need to perform manual steps for shutting down your system, invoking OPatch, applying SQL, and other such best-practice steps in the patching procedure.

SQL Performance Analyzer: Overview

- Targeted users: DBAs, QAs, application developers
- Helps predict the impact of system changes on SQL workload response time
- Builds different versions of SQL workload performance (that is, SQL execution plans and execution statistics)
- Executes SQL serially (concurrency not honored)
- Analyzes performance differences
- Offers fine-grained performance analysis on individual SQL
- Is integrated with SQL Tuning Advisor to tune regressions



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Oracle Real Application Testing option includes SQL Performance Analyzer, which gives you an exact and accurate assessment of the impact of change on the SQL statements that make up the workload. SQL Performance Analyzer helps you forecast the impact of a potential change on the performance of a SQL query workload. This capability provides DBAs with detailed information about the performance of SQL statements, such as before-and-after execution statistics, and statements with performance improvement or degradation. This enables you (for example) to make changes in a test environment to determine whether the workload performance will be improved through a database upgrade.

SQL Performance Analyzer: Use Cases

SQL Performance Analyzer is beneficial in the following use cases:

- Database upgrades
- Implementation of tuning recommendations
- Schema changes
- Statistics gathering
- Database parameter changes
- OS and hardware changes



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SQL Performance Analyzer can be used to predict and prevent potential performance problems for any database environment change that affects the structure of the SQL execution plans. The changes can include (but are not limited to) any of the following:

- Database upgrades
- Implementation of tuning recommendations
- Schema changes
- Statistics gathering
- Database parameter changes
- OS and hardware changes

You can use SQL Performance Analyzer to predict SQL performance changes that result from changes for even the most complex environments. As applications evolve through the development life cycle, database application developers can test changes to schemas, database objects, and rewritten applications to mitigate any potential performance impact.

SQL Performance Analyzer also enables the comparison of SQL performance statistics.

Using SQL Performance Analyzer



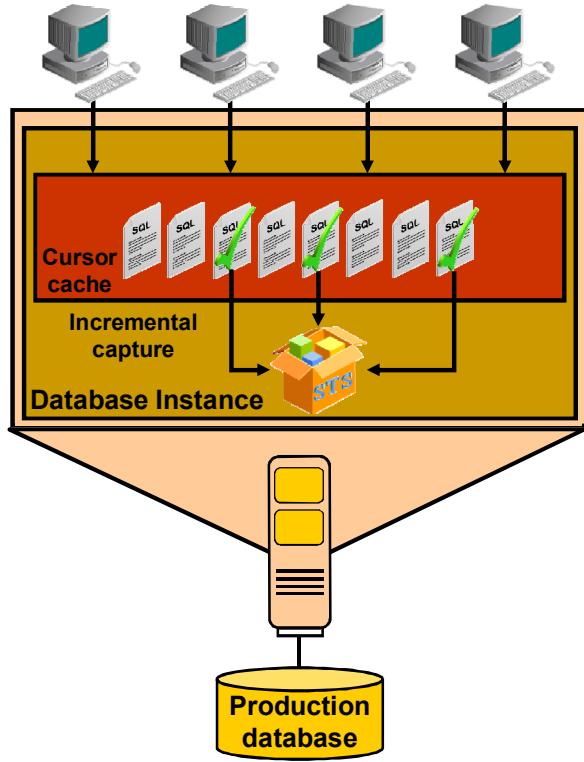
1. Capture SQL workload on production.
2. Transport the SQL workload to a test system.
3. Build “before-change” performance data.
4. Make changes.
5. Build “after-change” performance data.
6. Compare results from steps 3 and 5.
7. Tune regressed SQL.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

1. **Gather SQL:** In this phase, you collect the set of SQL statements that represent your SQL workload on the production system.
2. **Transport:** You must transport the resultant workload to the test system. The STS is exported from the production system and the STS is imported into the test system.
3. **Compute “before-version” performance:** Before any changes take place, you execute the SQL statements, collecting baseline information that is needed to assess the impact that a future change might have on the performance of the workload.
4. **Make a change:** After you have the before-version data, you can implement your planned change and start viewing the impact on performance.
5. **Compute “after-version” performance:** This step takes place after the change is made in the database environment. Each statement of the SQL workload runs under a mock execution (collecting statistics only), collecting the same information as captured in step 3.
6. **Compare and analyze SQL Performance:** After you have both versions of the SQL workload performance data, you can carry out the performance analysis by comparing the after-version data with the before-version data.
7. **Tune regressed SQL:** At this stage, you have identified exactly which SQL statements may cause performance problems when the database change is made. Here, you can use any of the database tools to tune the system. After implementing any tuning action, you should repeat the process to create a new after-version and analyze the performance differences to ensure that the new performance is acceptable.

Step 1: Capture SQL Workload



- SQL Tuning Set (STS) is used to store SQL workload. It includes:
 - SQL Text
 - Bind variables
 - Execution plans
 - Execution statistics
- Incremental capture is used to populate STS from cursor cache over a period of time.
- STS's filtering and ranking capabilities filter out undesirable SQL.

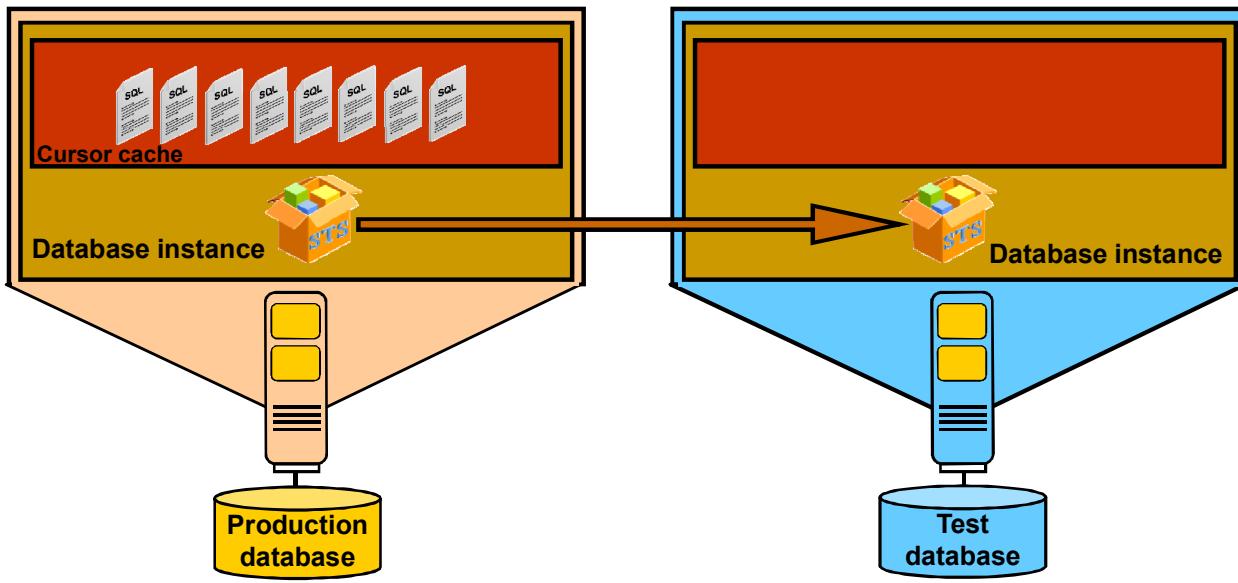
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The first step to using SQL Performance Analyzer is to capture the SQL statements that represent your workload.

You can use SQL Tuning Sets or Automatic Workload Repository (AWR) to capture the information to transport. Because AWR essentially captures high-load SQL statements, you should consider modifying the default AWR snapshot settings and captured Top SQL to ensure that AWR captures the maximum number of SQL statements. This ensures more complete SQL workload capture.

Step 2: Transport to a Test System



- Copy SQL Tuning Set to staging table (“pack”).
- Transport staging table to test system (data pump, DB link, and so on).
- Copy SQL Tuning Set from staging table (“unpack”).

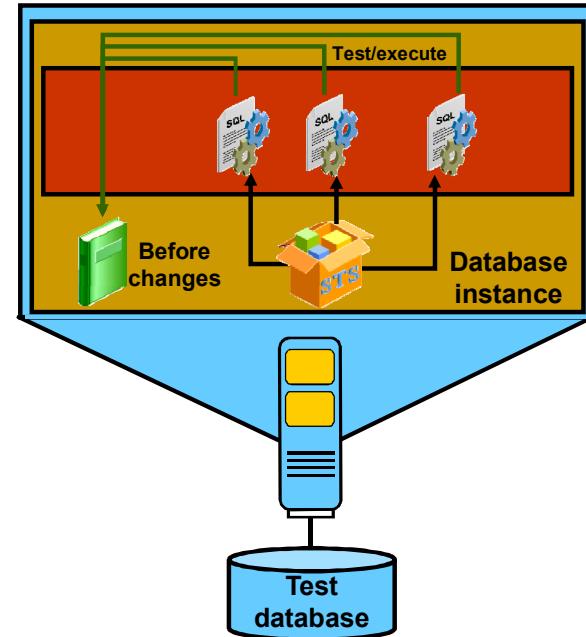
ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The second step is to transport these SQL statements to a similar system that is being tested. Here, STS can be exported from production and then imported into a test system.

Step 3: Build Before Change Performance Data

- Before change, SQL performance version is the SQL workload performance baseline.
- SQL performance = execution plans + execution statistics
- Test/execute SQL in STS:
 - Produce execution plans and statistics.
 - Execute SQL serially (no concurrency).
 - Every SQL is executed at least twice.
 - Skip DDL/DML effects.
- Explain plan SQL in STS generates only SQL plans.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The third step is to capture a baseline of the test system performance consisting of the execution plan and execution statistics.

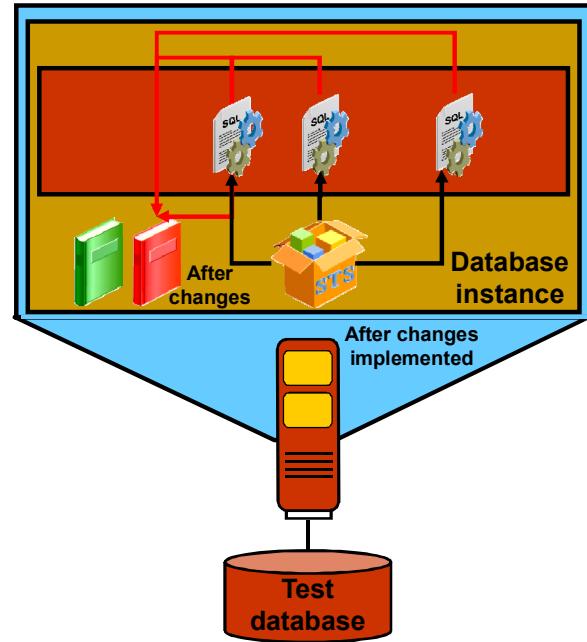
The information collected in this stage represents a snapshot of the current state of the system workload. The performance data includes:

- Execution plans (for example, generated by explain plan)
- Execution statistics (for example, includes elapsed time, buffer gets, disk reads, and rows processed)

In Oracle Database 11g, Release 2, each SQL statement is executed at least twice, for as many times as possible until the execution times out (up to a maximum of 10 times). The first execution is used to warm the buffer cache. All subsequent executions are then used to calculate the run-time execution statistics for the SQL statement based on their averages.

Step 4: Implement Planned Change and Step 5: Build After-Change Performance Data

- Manually implement the planned change:
 - Database upgrade
 - Implementation of tuning recommendations
 - Schema changes
 - Statistics gathering
 - Database parameter changes
 - OS and hardware changes
- Reexecute SQL after change:
 - Test/execute SQL in STS to generate SQL execution plans and statistics.
 - Explain plan SQL in STS to generate SQL plans.



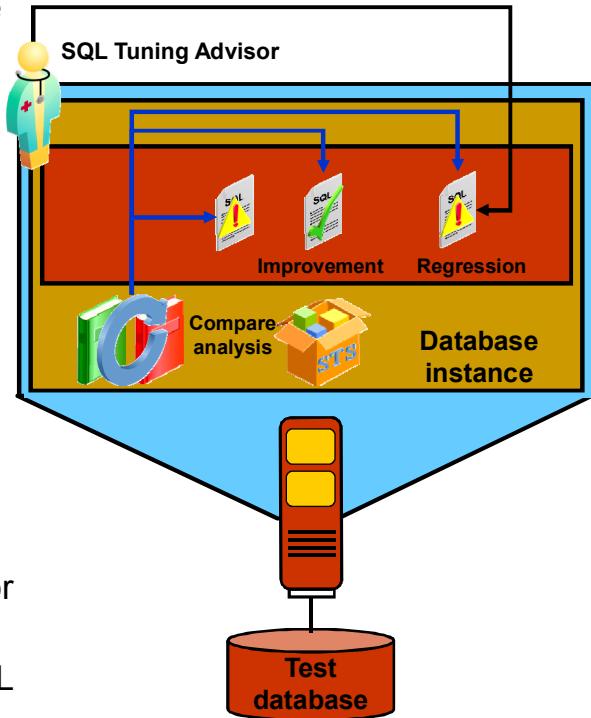
ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The fourth step is to make the changes to the test system. Rerun the SQL statements to assess the impact of the changes on the SQL performance as step 5.

Step 6: Compare and Analyze Performance and Step 7: Tune Regressed SQL

- Rely on user-specified metric to compare SQL performance:
 - ELAPSED_TIME, BUFFER_GETS, DISK_READS, ...
- Calculate impact of change on individual SQLs and SQL workload:
 - Overall impact on workload
 - Net SQL impact on workload
- Use SQL execution frequency to define a weight of importance.
- Detect improvements, regressions, and unchanged performance.
- Detect changes in execution plans.
- Recommend running SQL Tuning Advisor to tune regressed SQLs.
- Analysis results can be used to seed SQL Plan Management baselines.



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The comparison is based on the execution statistics, such as elapsed time, CPU time, and buffer gets.

Enterprise Manager provides the tools to make a full comparison of performance data, including execution statistics such as elapsed time, CPU time, and buffer gets. If the SQL performance has regressed in some of the cases, you must then run SQL Tuning Advisor to tune the SQL statements—either immediately or at a scheduled time. As with any tuning strategy, it is recommended that only one change be implemented at a time and retested before making further changes.

You can use SQL Tuning Advisor or Access Advisor against the identified statements and then implement those recommendations. Alternatively, you can seed SQL Plan Management (SPM) with plans captured in step 3 to guarantee that the plans remain the same.

Quiz

Which of the following does SQL Performance Analyzer perform?

- a. Tunes regressions
- b. Provides before-and-after execution statistics
- c. Executes SQL statements serially
- d. Builds different versions of SQL workload performance



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Answer: b, c, d

Accessing SQL Performance Analyzer

- Use Enterprise Manager.
- Use the DBMS_SQLPA package.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can access SQL Performance Analyzer through Enterprise Manager or by using the DBMS_SQLPA package.

Using Enterprise Manager to Access SQL Performance Analyzer

- Access SQL Performance Analyzer on the Software and Support tab.
- Select one of the five types of workflows:
 - Upgrade from 9*i* or 10.1
 - Upgrade from 10.2 or 11*g*
 - Parameter Change
 - Exadata Simulation
 - Guided Workflow
- Tune regressing statements by invoking SQL Tuning Advisor.
- Prevent regressions by using SQL plan baselines.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You access SQL Performance Analyzer from the “Software and Support” tab of Database Control. Alternatively, select Database Instance > Advisor Central > Advisors > SQL Performance Analyzer.

SQL Performance Analyzer offers three workflows for you to test different scenarios:

- **Upgrade from 9*i* or 10.1:** Test and analyze the effects of database upgrade from 9*i* or 10.1 on SQL Tuning Set performance.
- **Upgrade from 10.2 or 11*g*:** Test and analyze the effects of database upgrade from 10.2 or 11*g* on SQL Tuning Set performance.
- **Parameter Change:** Test and compare an initialization parameter change on SQL Tuning Set performance. A SQL Performance Analyzer task is created and an initial trial run is performed with the parameter set to the base value. A second trial run is performed with the parameter set to the changed value. A replay trial comparison report is then run for the two trials.
- **Exadata Simulation:** Simulate the effects of an Exadata Storage Server installation on SQL Tuning Set performance.
- **Guided Workflow:** Create a SQL Performance Analyzer task and execute custom experiments by using manually created replay trials.

You can directly tune all regressing statements by invoking SQL Tuning Advisor. Instead of using SQL Tuning Advisor to tune your regressing statements, you can also prevent regressions by using the SQL plan baselines.

SQL Performance Analyzer: PL/SQL Example

- Create the tuning task:

```
exec :tname:= dbms_sqlpa.create_analysis_task( -  
    sqlset_name => 'MYSTS', task_name => 'MYSPL');
```

- Execute the task to build the before-change performance data:

```
exec dbms_sqlpa.execute_analysis_task(task_name => :tname, -  
    execution_type => 'TEST EXECUTE', execution_name => 'before');
```

- Produce the before-change report:

```
SELECT dbms_sqlpa.report_analysis_task(task_name => :tname,  
    type=>'text', section=>'summary') FROM dual;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The example in the slide shows you how to use the DBMS_SQLPA package to invoke SQL Performance Analyzer to access the SQL performance impact of some changes. You could easily adapt this example to run your own analysis.

1. Create the tuning task to run SQL Performance Analyzer.
2. Execute the task once to build the before-change performance data, and produce the before-change report (special settings for report: set long 100000, longchunksize 100000, and linesize 90). With this call, you can specify various parameters, some of which are:
 - Set the EXECUTION_TYPE parameter as follows: EXPLAIN_PLAN to generate explain plans for all SQL statements in the SQL workload. TEST_EXECUTE to execute all SQL statements in the SQL workload. The procedure executes only the query part of the DML statements to prevent side-effects to the database or user data. When TEST_EXECUTE is specified, the procedure generates execution plans and execution statistics. COMPARE [PERFORMANCE] to analyze and compare two versions of SQL performance data. CONVERT_SQLSET to read the statistics captured in a SQL Tuning Set and model them as a task execution.
 - Specify execution parameters by using the execution_params parameter that needs to be specified as dbms_advisor.arglist(name,value,...). The time_limit parameter specifies the global time limit to process all SQL statements in a SQL Tuning Set before timing out. The local_time_limit parameter specifies the time limit to process each SQL statement in a SQL Tuning Set before timing out.

SQL Performance Analyzer: PL/SQL Example

After making your changes:

- Create the after-change performance data:

```
EXEC dbms_sqlpa.execute_analysis_task(task_name => :tname, -  
execution_type => 'TEST EXECUTE', execution_name => 'after');
```

- Generate the after-change report:

```
SELECT dbms_sqlpa.report_analysis_task(task_name => :tname,  
type=>'text', section=>'summary') FROM dual;
```

- Compare the task executions:

```
EXEC dbms_sqlpa.execute_analysis_task(task_name => :tname,  
execution_type => 'COMPARE PERFORMANCE');
```

- Generate the analysis report:

```
SELECT dbms_sqlpa.report_analysis_task(task_name => :tname,  
type=>'text', section=>'summary') FROM dual;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

3. Make your changes.
4. Execute the task again after making the changes, and generate the after-changes report.
5. Compare the two executions and generate the analysis report. Using different execution parameters, you can execute the following command:

```
EXEC dbms_sqlpa.execute_analysis_task(task_name => :tname,  
execution_type => 'COMPARE PERFORMANCE');
```

Note: For more information about the DBMS_SQLPA package, see the *Oracle Database PL/SQL Packages and Types Reference Guide*.

Tuning Regressed SQL Statements

To tune regressed SQL statements reported by SQL Performance Analyzer, create a SQL tuning task for the SQL Performance Analyzer execution by using the DBMS_SQLTUNE.CREATE_TUNING_TASK function:

```
BEGIN  
DBMS_SQLTUNE.CREATE_TUNING_TASK(  
spa_task_name => 'MYSQA',  
spa_compare_exec => 'MYCOMPEEXEC');  
END;  
/
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

After reviewing the SQL Performance Analyzer report, you should tune any regressed SQL statements that are identified after comparing the SQL performance. If there are a large number of SQL statements that appear to have regressed, try to identify the root cause and make system-level changes to rectify the problem.

If only a few SQL statements have regressed, consider using the SQL Tuning Advisor to implement a point solution for them. Create a SQL tuning task for the SQL Performance Analyzer execution by using the DBMS_SQLTUNE.CREATE_TUNING_TASK function. The CREATE_TUNING_TASK function has the following parameters:

- SPA_TASK_NAME: Name of the SQL Performance Analyzer task
- SPA_TASK_OWNER: Owner of the specified SQL Performance Analyzer task. If unspecified, this parameter will default to the current user.
- SPA_COMPARE_EXEC: Execution name of the compare performance trial for the specified SQL Performance Analyzer task. If unspecified, this parameter defaults to the most recent execution of the COMPARE PERFORMANCE type for the given SQL Performance Analyzer task.

After tuning the regressed SQL statements, test your changes by using SQL Performance Analyzer. Run a new SQL trial on the test system, followed by a second comparison (between this new SQL trial and the first SQL trial) to validate your results. After SQL Performance Analyzer shows that performance has stabilized, you can implement the changes.

Testing Database Upgrades: Oracle9*i* Database and Oracle Database 10g Release 1

- SQL Performance Analyzer supports testing database upgrades of Oracle9*i* and Oracle Database 10g Release 1, to Oracle Database 10g Release 2 and later releases.
- Execute the SQL tuning set on the upgraded database remotely over a database link.
- The production system which you are upgrading *from* should be running Oracle9*i* or Oracle Database 10g Release 1.
- The test system which you are upgrading *to* should be running Oracle Database 10g Release 2 (10.2.0.2) or a newer release.
- Set up a separate system for SQL Performance Analyzer: Oracle Database 11g Release 1 (11.1.0.7) or a later release.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SQL Performance Analyzer supports testing database upgrades of Oracle9*i* and Oracle Database 10g Release 1 to Oracle Database 10g Release 2 and later releases by executing the SQL tuning set on the upgraded database remotely over a database link. Because SQL Performance Analyzer only accepts a set of SQL statements stored in a SQL tuning set as its input source, and SQL tuning sets are not supported in Oracle9*i*, a SQL tuning set must be constructed so that it can be used as an input source for SQL Performance Analyzer if you are upgrading from Oracle9*i*.

The production system which you are upgrading from should be running Oracle9*i* or Oracle Database 10g Release 1. The test system that you are upgrading to should be running Oracle Database 10g Release 2 (10.2.0.2) or a newer release. If you are upgrading to Oracle Database 10g Release 10.2.0.2, 10.2.0.3, or 10.2.0.4, you will also need to install a one-off patch before proceeding.

Set up a separate system for SQL Performance Analyzer running Oracle Database 11g Release 1 (11.1.0.7) or a later release. Use this system to build a SQL tuning set and to run SQL Performance Analyzer. You do not need your production data or schema on this system, because the SQL tuning set will be built using statistics stored in the SQL trace files from the production system. SQL Performance Analyzer tasks will be executed remotely on the test system to generate the execution plan and statistics for the SQL trial over a database link that you specify. The database link must be a public database link that connects to a user with the EXECUTE privilege for the DBMS_SQLPA package and the ADVISOR privilege on the test system. You should also drop any existing PLAN_TABLE from the user's schema on the test system.

Testing Database Upgrades: Oracle9*i* Database and Oracle Database 10g Release 1

To use SQL Performance Analyzer in a database upgrade from Oracle9*i* or Oracle Database 10g Release 1 to a newer release, perform the following steps:

1. Enable the SQL Trace facility on the production system.
2. On the production system, create a mapping table.
3. Move the SQL trace files and the mapping table from the production system to the SQL Performance Analyzer system.
4. On the SQL Performance Analyzer system, construct a SQL tuning set by using the SQL trace files.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To use SQL Performance Analyzer in a database upgrade from Oracle9*i* or Oracle Database 10g Release 1 to a later release, perform the following steps:

1. Enable the SQL Trace facility on the production system. Consider enabling SQL Trace for only a subset of the sessions, for as long as required, to capture all important SQL statements at least once.
2. On the production system, create a mapping table that will be used to convert the user and object identifier numbers in the SQL trace files to their string equivalents.
3. Move the SQL trace files and the mapping table from the production system to the SQL Performance Analyzer system.
4. On the SQL Performance Analyzer system, construct a SQL tuning set by using the SQL trace files. The SQL tuning set will contain the SQL statements captured in the SQL trace files, along with their relevant execution context and statistics.

Testing Database Upgrades: Oracle9i Database and Oracle Database 10g Release 1

5. On the SQL Performance Analyzer system:
 - Use SQL Performance Analyzer to create a SQL Performance Analyzer task and convert the contents in the SQL tuning set into a preupgrade SQL trial that will be used as a baseline for comparison.
 - Remotely test execute the SQL statements on the test system over a database link to build a postupgrade SQL trial.
6. Compare SQL performance and fix regressed SQL statements.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

5. On the SQL Performance Analyzer system, use SQL Performance Analyzer to create a SQL Performance Analyzer task and convert the contents in the SQL tuning set into a preupgrade SQL trial that will be used as a baseline for comparison. Remotely test execute the SQL statements on the test system over a database link to build a postupgrade SQL trial. You can access the SQL Performance Analyzer through Enterprise Manager or by using the DBMS_SQLPA package.
6. Compare SQL performance and fix any regressed SQL statements.

Repeat the process of executing the SQL tuning set and comparing its performance to a previous execution to test any changes made until you are satisfied with the results.

Testing Database Upgrades: Oracle Database 10g Release 2 and Later Releases

- SQL Performance Analyzer supports testing database upgrades of Oracle Database 10g Release 2 or a later release to any later release.
- Capture a SQL tuning set on the production system, then execute it twice remotely over a database link on a test system.
- The production system which you are upgrading *from* should be running Oracle Database 10g Release 2 or a later release.
- The test system which you are upgrading *to initially* should be running the same release as the production system.
- Set up a separate system for running SQL Performance Analyzer: Oracle Database 11g Release 1 (11.1.0.7) or a later release.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use SQL Performance Analyzer to test the impact on SQL response time of a database upgrade from Oracle Database 10g Release 2 or a later release to any later release by capturing a SQL tuning set on the production system, then executing it twice remotely over a database link on a test system—first to create a prechange SQL trial, and then to create a postchange SQL trial.

The production system that you are upgrading *from* should be running Oracle Database 10g Release 2 or a later release. Initially, the *test system* should also be running the same release. To ensure that the analysis made by SQL Performance Analyzer is accurate, the test system should contain an exact copy of the production data found on the production system. The hardware configuration on the test system should be as similar to the production system as possible.

Set up a separate system for SQL Performance Analyzer running Oracle Database 11g Release 1 (11.1.0.7) or a later release. Use this system to build a SQL tuning set and to run SQL Performance Analyzer. You do not need your production data or schema on this system, because the SQL tuning set will be built using statistics stored in the SQL trace files from the production system. SQL Performance Analyzer tasks will be executed remotely on the test system to generate the execution plan and statistics for the SQL trial over a database link that you specify. The database link must be a public database link that connects to a user with the EXECUTE privilege for the DBMS_SQLPA package and the ADVISOR privilege on the test system. You should also drop any existing PLAN_TABLE from the user's schema on the test system.

Testing Database Upgrades: Oracle Database 10g Release 2 and Later Releases

Perform the following steps to use SQL Performance Analyzer in a database upgrade from Oracle Database 10g Release 2 and later releases to a newer release:

1. On the production system, capture the SQL workload that you intend to analyze and store it in a SQL tuning set.
2. Set up the test system so that it matches the production environment as closely as possible.
3. Transport the SQL tuning set to the SQL Performance Analyzer system.
4. On the SQL Performance Analyzer system, create a SQL Performance Analyzer task using the SQL tuning set as its input source.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To use SQL Performance Analyzer in a database upgrade from Oracle Database 10g Release 2 and later releases to a newer release, perform the following steps:

1. On the production system, capture the SQL workload that you intend to analyze and store it in a SQL tuning set.
2. Set up the test system so that it matches the production environment as closely as possible.
3. Transport the SQL tuning set to the SQL Performance Analyzer system
4. On the SQL Performance Analyzer system, create a SQL Performance Analyzer task using the SQL tuning set as its input source. Remotely test execute the SQL statements in the SQL tuning set on the test system over a database link to build a preupgrade SQL trial that will be used as a baseline for comparison.

Testing Database Upgrades: Oracle Database 10g Release 2 and Later Releases

5. Upgrade the test system.
6. Remotely test execute the SQL statements a second time on the upgraded test system over a database link to build a postupgrade SQL trial.
7. Compare SQL performance and fix regressed SQL statements.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

5. Upgrade the test system.
6. Remotely test execute the SQL statements a second time on the upgraded test system over a database link to build a postupgrade SQL trial.
7. Compare SQL performance and fix any regressed SQL statements.

Repeat the process of executing the SQL tuning set and comparing its performance to a previous execution to test any changes made until you are satisfied with the results.

SQL Performance Analyzer: Data Dictionary Views

- Modified views in Oracle Database 11g:
 - DBA{USER}_ADVISOR_TASKS: Displays details about the analysis task
 - DBA{USER}_ADVISOR_FINDINGS: Displays analysis findings
- New views in Oracle Database 11g:
 - DBA{USER}_ADVISOR_EXECUTIONS: Lists metadata information for task execution
 - DBA{USER}_ADVISOR_SQLPLANS: Displays the list of SQL execution plans
 - DBA{USER}_ADVISOR_SQLSTATS: Displays the list of SQL compilation and execution statistics



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Modified views in Oracle Database 11g:

- DBA{USER}_ADVISOR_TASKS: Displays details about the advisor task created to perform an impact analysis of a system environment change
- DBA{USER}_ADVISOR_FINDINGS: Displays analysis findings. The advisor generates four types of findings: performance regression, symptoms, errors, and informative messages.

New views in Oracle Database 11g:

- DBA{USER}_ADVISOR_EXECUTIONS: Lists metadata information for a task execution. SQL Performance Analyzer creates a minimum of three executions to perform a change impact analysis on a SQL workload: one execution to collect performance data for the before-change version of the workload, the second execution to collect data for the after-change version of the workload, and a final execution to perform the actual analysis.
- DBA{USER}_ADVISOR_SQLPLANS: Displays the list of all SQL execution plans (or those owned by the current user)
- DBA{USER}_ADVISOR_SQLSTATS: Displays the list of SQL compilation and execution statistics (or those owned by the current user)

Summary

In this lesson, you should have learned how to:

- Identify the benefits of using SQL Performance Analyzer
- Describe the SQL Performance Analyzer workflow phases
- Use SQL Performance Analyzer to determine performance gains following a database change



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

SQL Plan Management

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

- Set up SQL Plan Management
- Set up various SQL Plan Management scenarios
- Migrate stored outlines to SQL Plan baselines



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SQL Plan Management: Overview

- SQL Plan Management is automatically controlled SQL plan evolution.
- Optimizer automatically manages SQL plan baselines.
 - Only known and verified plans are used.
- Plan changes are automatically verified.
 - Only comparable or better plans are used going forward.
- Can preseed critical SQL with SQL Tuning Set (STS) from SQL Performance Analyzer
- Main benefit is the performance stability of the system through the avoidance of plan regressions.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

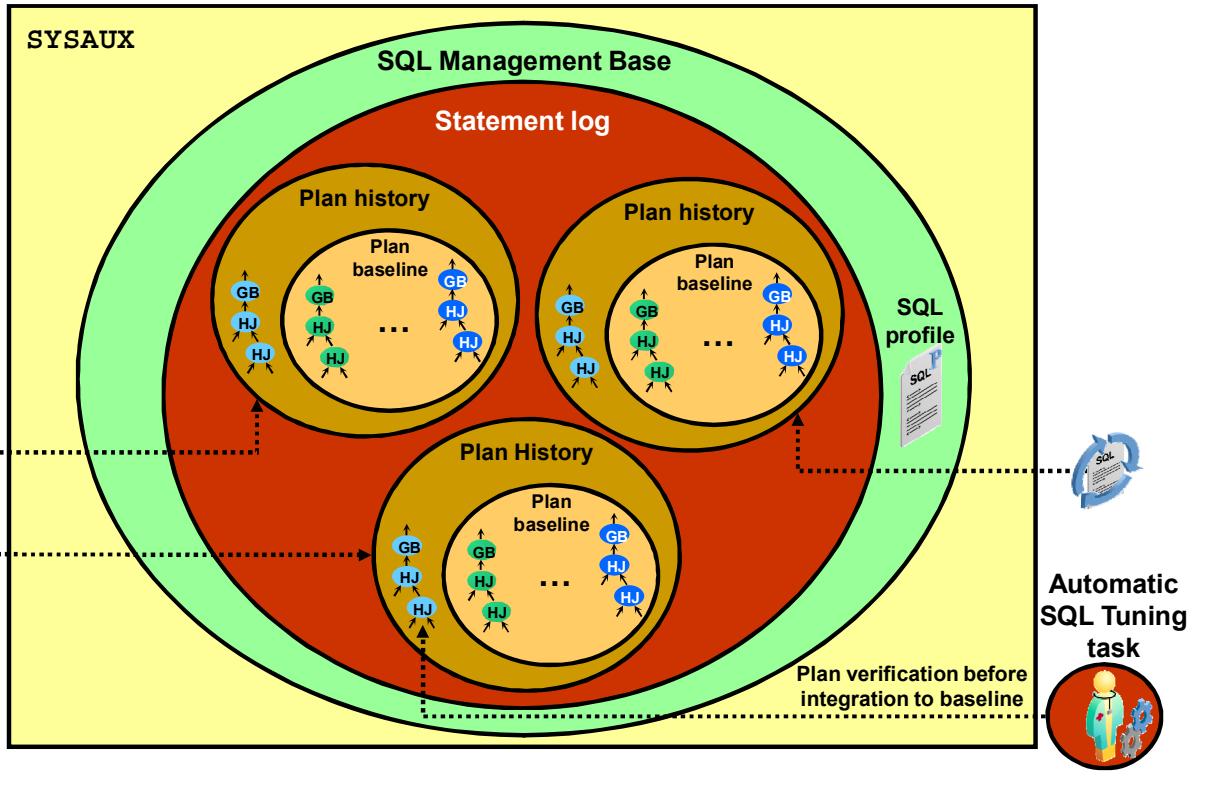
Potential performance risk occurs when the SQL execution plan changes for a SQL statement. A SQL plan change can occur due to a variety of reasons such as optimizer version, optimizer statistics, optimizer parameters, schema definitions, system settings, and SQL profile creation.

Various plan control techniques (such as stored outlines and SQL profiles) have been introduced in past versions of Oracle Database to address performance regression due to plan changes. However, these techniques are reactive processes that require manual intervention.

SQL Plan Management is a new feature introduced with Oracle Database 11g that enables the system to automatically control SQL plan evolution by maintaining *SQL plan baselines*. With this feature enabled, a newly generated SQL plan can integrate a SQL plan baseline only if it has been proven that doing so will not result in performance regression. During execution of a SQL statement, only a plan that is part of the corresponding SQL plan baseline can be used. As described later in this lesson, SQL plan baselines can be automatically loaded or can be seeded using SQL Tuning Sets. Various scenarios are covered later in this lesson.

The main benefit of the SQL Plan Management feature is the performance stability of the system through the avoidance of plan regressions. In addition, it saves the DBA time that is often spent in identifying and analyzing SQL performance regression and finding workable solutions.

SQL Plan Baseline: Architecture



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

The SQL Plan Management (SPM) feature introduces necessary infrastructure and services in support of plan maintenance and performance verification of new plans.

For SQL statements that are executed more than once, the optimizer maintains a history of plans for individual SQL statements. The optimizer recognizes a repeatable SQL statement by maintaining a statement log. A SQL statement is recognized as repeatable when it is parsed or executed again after it has been logged. After a SQL statement is recognized as repeatable, various plans generated by the optimizer are maintained as a plan history containing relevant information (such as SQL text, outline, bind variables, and compilation environment) that is used by the optimizer to reproduce an execution plan.

As an alternative or complement to the automatic recognition of repeatable SQL statements and the creation of their plan history, manual seeding of plans for a set of SQL statements is also supported.

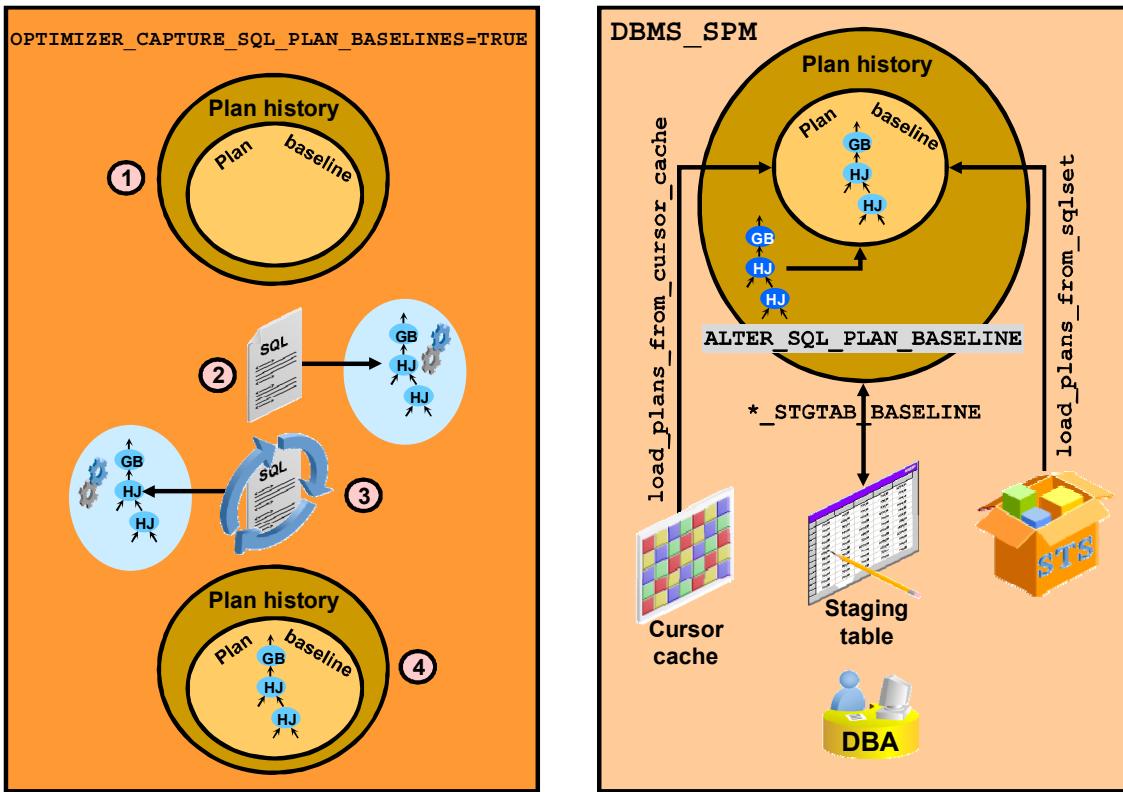
A plan history contains different plans generated by the optimizer for a SQL statement over time. However, only some of the plans in the plan history may be accepted for use. For example, a new plan generated by the optimizer is not normally used until it has been verified not to cause a performance regression. Plan verification is done “out of the box” as part of Automatic SQL Tuning running as an automated task in a maintenance window.

An Automatic SQL Tuning task targets only high-load SQL statements. For them, it automatically implements actions such as making a successfully verified plan an accepted plan. A set of acceptable plans constitutes a SQL plan baseline. The very first plan generated for a SQL statement is obviously acceptable for use; therefore, it forms the original plan baseline. Any new plans subsequently found by the optimizer are part of the plan history but not part of the plan baseline initially.

The statement log, plan history, and plan baselines are stored in the SQL Management Base (SMB), which also contains SQL profiles. The SMB is part of the database dictionary and is stored in the SYSAUX tablespace. The SMB has automatic space management (for example, periodic purging of unused plans). You can configure the SMB to change the plan retention policy and set space size limits.

Note: With Oracle Database 11g, if the database instance is up but the SYSAUX tablespace is OFFLINE, the optimizer is unable to access SQL management objects. This can affect performance on some of the SQL workload.

Loading SQL Plan Baselines



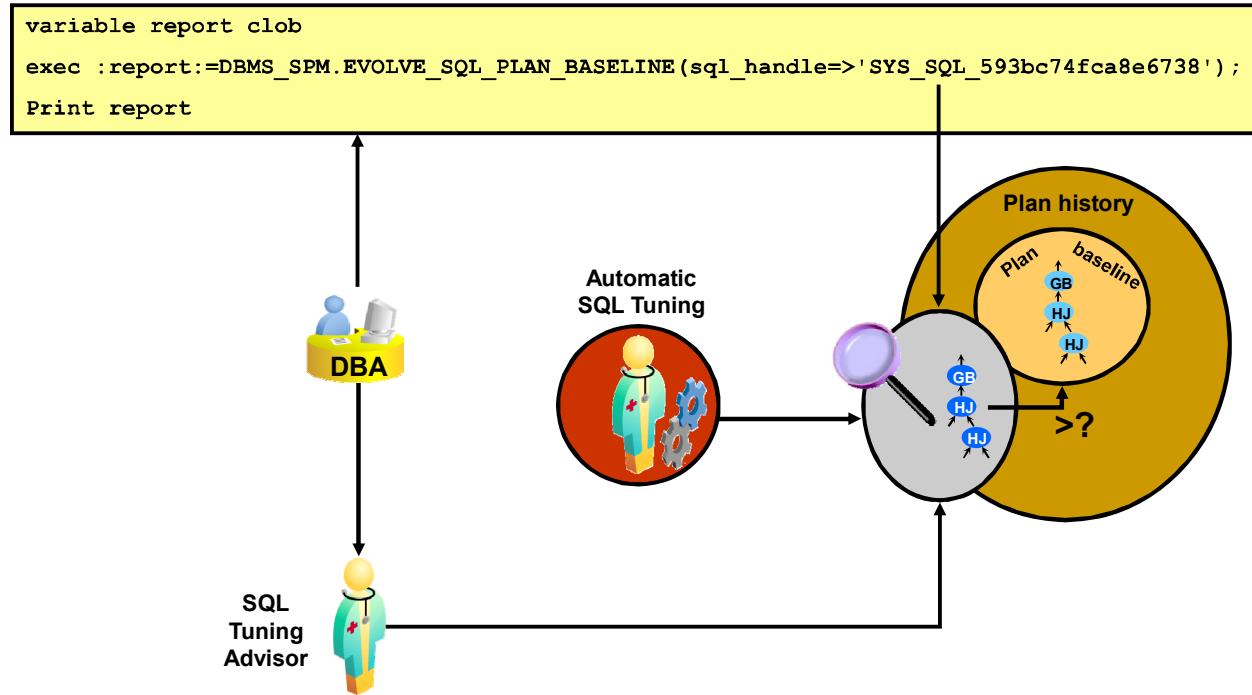
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can load SQL plan baselines as follows:

- On the fly capture:** Uses automatic plan capture by setting the initialization parameter `OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES` to `TRUE`. This parameter is set to `FALSE` by default. Setting it to `TRUE` turns on automatic recognition of repeatable SQL statements and automatic creation of plan history for such statements. This is illustrated in the left graphic in the slide, where you can see the first generated SQL plan automatically integrated into the original SQL plan baseline.
- Bulk loading:** Uses the `DBMS_SPM` package, which enables you to manually manage SQL plan baselines. With this package, you can load SQL plans into a SQL plan baseline directly from the cursor cache or from an existing SQL Tuning Set (STS). For a SQL statement to be loaded into a SQL plan baseline from an STS, the SQL statement needs to store its SQL plan in the STS. `DBMS_SPM` enables you to change the status of a baseline plan from accepted to not accepted (and from not accepted to accepted). It also enables you to export baseline plans from a staging table, which can then be used to load SQL plan baselines on other databases.
- Migrate from stored outlines:** Use the `DBMS_SPM.MIGRATE_STORED_OUTLINE` function to migrate stored outlines for one or more SQL statements to plan baselines in the SQL management base (SMB). Additional information is provided later in this lesson.

Evolving SQL Plan Baselines



ORACLE

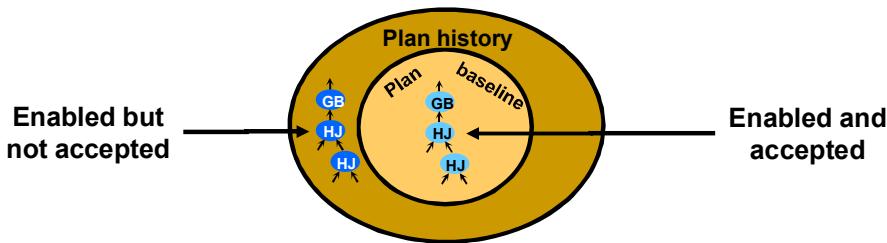
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

During the SQL plan baseline evolution phase, Oracle Database routinely evaluates the performance of new plans and integrates plans with better performance into SQL plan baselines. When the optimizer finds a new plan for a SQL statement, the plan is added to the plan history as a nonaccepted plan. The plan is then verified for performance relative to the SQL plan baseline performance. When it is verified that a nonaccepted plan does not cause a performance regression (either manually or automatically), the plan is changed to an accepted plan and integrated into the SQL plan baseline. Successful verification of a nonaccepted plan consists of comparing its performance to that of one plan selected from the SQL plan baseline and ensuring that it delivers better performance.

There are two ways to evolve SQL plan baselines:

- By using the `DBMS_SPM.EVOLVE_SQL_PLAN_BASELINE` function. An invocation example is shown in the slide. The function returns a report that tells you whether some of the existing history plans were moved to the plan baseline. You can also specify specific plans in the history to be tested.
- By running SQL Tuning Advisor: SQL plan baselines can be evolved by manually or automatically tuning SQL statements using SQL Tuning Advisor. When SQL Tuning Advisor finds a tuned plan and verifies its performance to be better than a plan chosen from the corresponding SQL plan baseline, it makes a recommendation to accept a SQL profile. When the SQL profile is accepted, the tuned plan is added to the corresponding SQL plan baseline.

Viewing Important Baseline SQL Plan Attributes



```
SELECT signature, sql_handle, sql_text, plan_name, origin, enabled,
       accepted, fixed, autopurge
  FROM dba_sql_plan_baselines;
```

SIGNATURE	SQL_HANDLE	SQL_TEXT	PLAN_NAME	ORIGIN	ENA	ACC	FIX	AUT
-----	-----	-----	-----	-----	-----	-----	-----	-----
8.062E+18	SYS_SQL_6fe2	select..	SYS_SQL_PLAN_1ea	AUTO-CAPTURE	YES	NO	NO	YES
8.062E+18	SYS_SQL_6fe2	select..	SYS_SQL_PLAN_4be	AUTO-CAPTURE	YES	YES	NO	YES
...								

```
exec :cnt := dbms_spm.alter_sql_plan_baseline(sql_handle => 'SYS_SQL_37e0168b0...3efe', -
                                                 plan_name      => 'SYS_SQL_PLAN_8dfc352f359901ea',
                                                 attribute_name => 'ENABLED', attribute_value => 'NO');
```

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can look at each plan's attributes by using the `DBA_SQL_PLAN_BASELINES` view, as shown in the slide. You can then use the `DBMS_SPM.ALTER_SQL_PLAN_BASELINE` function to change some of them. You can also remove plans or a complete plan history by using the `DBMS_SPM.DROP_SQL_PLAN_BASELINE` function. The example shown in the slide changes the `ENABLED` attribute of `SYS_SQL_PLAN_8DFC352F359901EA` to `NO`.

Additional information on the attributes is presented on the next page.

Note: The `DBA_SQL_PLAN_BASELINES` view contains additional columns that enable you to determine when each plan was last used and whether a plan will be automatically purged.

Important Baseline SQL Plan Attributes

- ORIGIN:
 - AUTO-CAPTURE: Automatically captured
 - MANUAL-LOAD: Manually evolved
 - MANUAL-SQLTUNE: Automatically evolved by SQL Tuning Advisor
 - AUTO-SQLTUNE: Automatically evolved by Automatic SQL Tuning
- ENABLED: Plan is enabled for use by optimizer
- ACCEPTED: Plan validated as a good plan
- FIXED: Optimizer considers only those plans

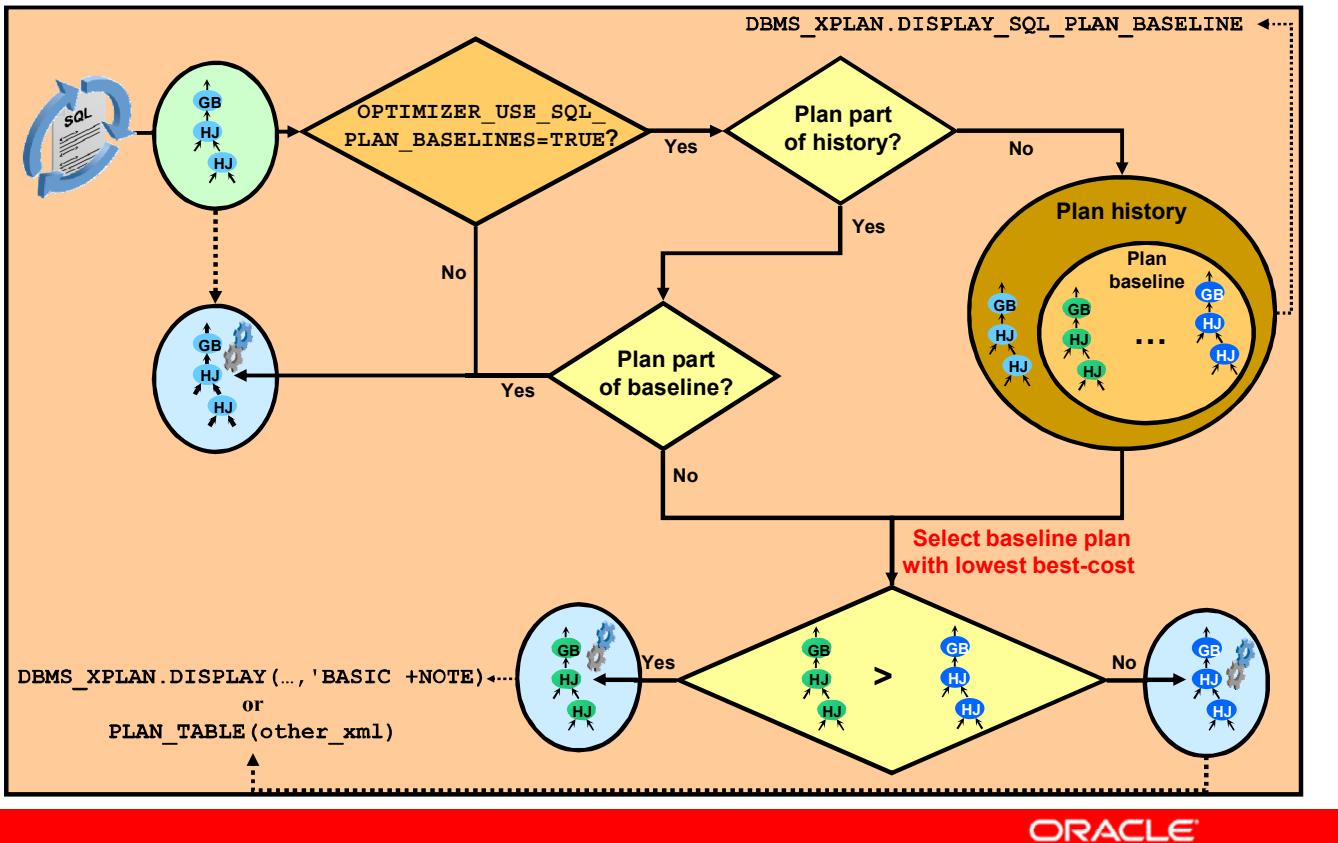


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When a plan enters the plan history, it is associated with a number of important attributes:

- SIGNATURE, SQL_HANDLE, SQL_TEXT, and PLAN_NAME are important identifiers for search operations.
- ORIGIN allows you to determine whether the plan was automatically captured (AUTO-CAPTURE), manually evolved (MANUAL-LOAD), automatically evolved by SQL Tuning Advisor (MANUAL-SQLTUNE), or automatically evolved by Automatic SQL Tuning (AUTO-SQLTUNE).
- ENABLED and ACCEPTED: The ENABLED attribute means that the plan is enabled for use by the optimizer. If ENABLED is not set, the plan is not considered. The ACCEPTED attribute means that the plan was validated as a good plan, either automatically by the system or manually when the user changes it to ACCEPTED. When a plan changes to ACCEPTED, it will become not ACCEPTED only when DBMS_SPM.ALTER_SQL_PLAN_BASELINE() is used to change its status. An ACCEPTED plan can be temporarily disabled by removing the ENABLED setting. A plan must be ENABLED and ACCEPTED for the optimizer to consider using it.
- FIXED means that the optimizer considers only those plans and not other plans. For example, if you have 10 baseline plans and three of them are marked FIXED, the optimizer uses only the best plan from these three, ignoring all the others. A SQL plan baseline is said to be FIXED if it contains at least one enabled fixed plan. If new plans are added to a fixed SQL plan baseline, these new plans cannot be used until they are manually declared as FIXED.

SQL Plan Selection



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you are using automatic plan capture, the first time that a SQL statement is recognized as repeatable, its best-cost plan is added to the corresponding SQL plan baseline. That plan is then used to execute the statement.

The optimizer uses a comparative plan selection policy when a plan baseline exists for a SQL statement and the OPTIMIZER_USE_SQL_PLAN_BASELINES initialization parameter is set to TRUE (default value). Each time a SQL statement is compiled, the optimizer first uses the traditional cost-based search method to build a best-cost plan. Then it tries to find a matching plan in the SQL plan baseline. If a match is found, it proceeds as usual. If no match is found, it first adds the new plan to the plan history, then costs each of the accepted plans in the SQL plan baseline, and picks the one with the lowest cost. The accepted plans are reproduced using the outline that is stored with each of them. So the effect of having a SQL plan baseline for a SQL statement is that the optimizer always selects one of the accepted plans in that SQL plan baseline.

With SQL Plan Management, the optimizer can produce a plan that could be either a best-cost plan or a baseline plan. This information is dumped in the OTHER_XML column of the PLAN_TABLE upon EXPLAIN PLAN.

In addition, you can use the new DBMS_XPLAIN.DISPLAY_SQL_PLAN_BASELINE function to display one or more execution plans for the specified `sql_handle` of a plan baseline. If `PLAN_NAME` is also specified, the corresponding execution plan is displayed.

Note: To preserve backward compatibility, if a stored outline for a SQL statement is active for the user session, the statement is compiled using the stored outline. In addition, a plan generated by the optimizer using a stored outline is not stored in the SMB even if automatic plan capture has been enabled for the session.

Quiz

In which of the following ways can SQL Plan baselines be loaded?

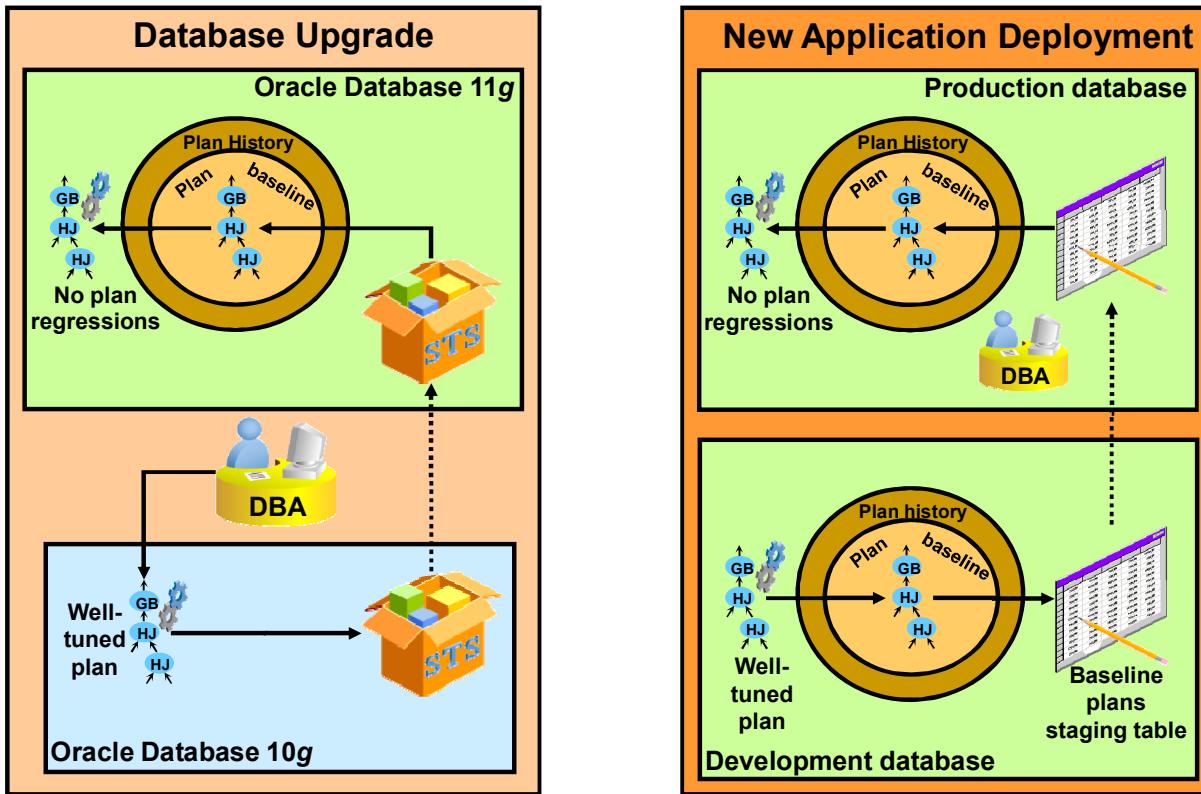
- a. Automatically by setting
OPTIMIZER_CAPTURE_SQL_PLAN_BASELINES to TRUE
- b. From SQL Performance Analyzer SQL Tuning Set (STS) by using the DBMS_SPM package
- c. From the cursor cache by using the DBMS_SPM package
- d. Automatically by default



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Answer: a, b, c

Possible SQL Plan Manageability Scenarios



ORACLE

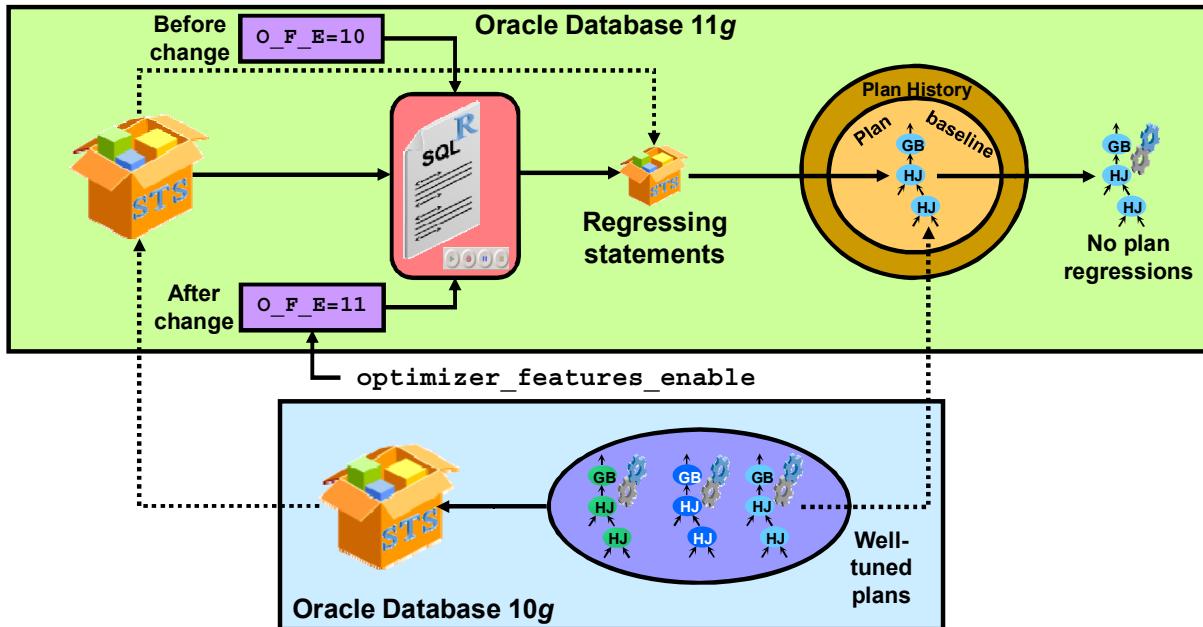
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

- **Database upgrade:** Bulk SQL plan loading is especially useful when the system is being upgraded from an earlier version to Oracle Database 11g. For this, you can capture plans for a SQL workload into a SQL Tuning Set (STS) before the upgrade, and then load these plans from the STS into the SQL plan baseline immediately after the upgrade. This strategy can minimize plan regressions resulting from the use of the new optimizer version.
- **New application deployment:** The deployment of a new application module means the introduction of new SQL statements into the system. The software vendor can ship the application software along with the appropriate SQL plan baselines for the new SQL being introduced. Because of the plan baselines, the new SQL statements will initially run with the plans that are known to give good performance under a standard test configuration. However, if the customer system configuration is very different from the test configuration, the plan baselines can be evolved over time to produce better performance.

In both scenarios, you can use the automatic SQL plan capture after manual loading to make sure that only better plans will be used for your applications in the future.

Note: In all scenarios in this lesson, assume that `OPTIMIZER_USE_SQL_PLAN_BASELINES` is set to TRUE.

SQL Performance Analyzer and SQL Plan Baseline Scenario



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

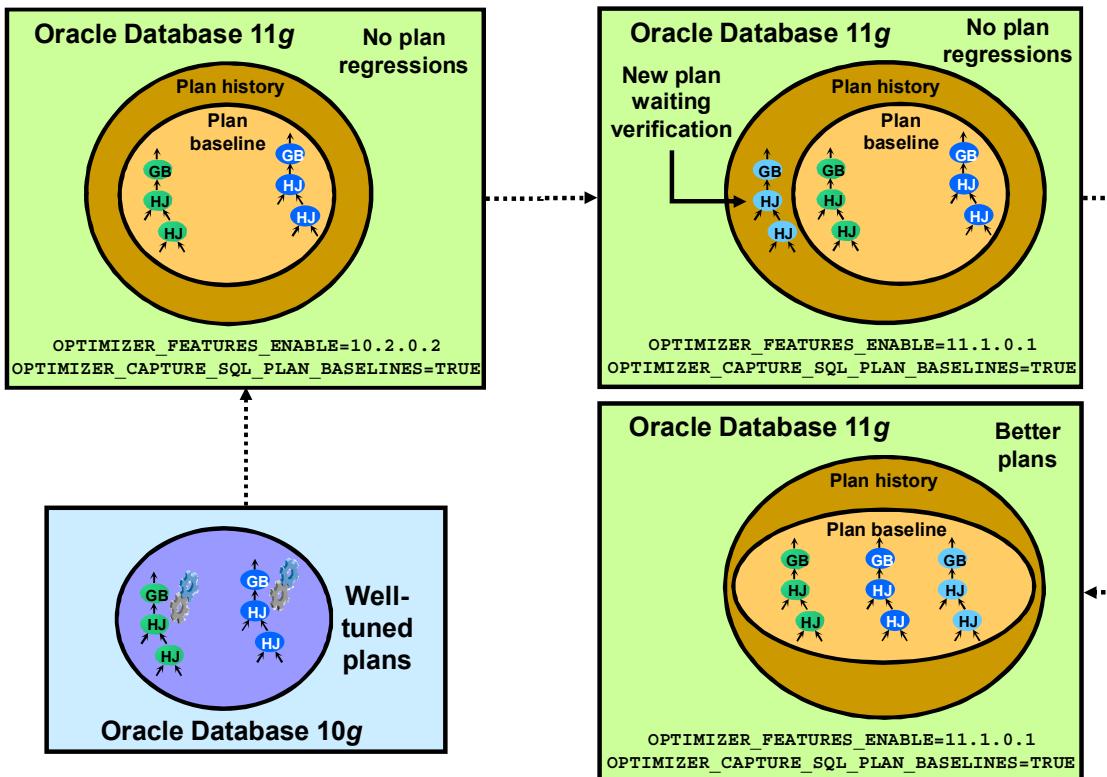
A variation of the first method described in the previous slide is through the use of SQL Performance Analyzer as follows:

1. You can capture pre-Oracle Database 11g plans in an STS and import them into Oracle Database 11g.
2. Then set the `OPTIMIZER_FEATURES_ENABLE` initialization parameter to 10.2 to make the optimizer behave as if this were a 10g Oracle database.
3. Next run SQL Performance Analyzer for the STS.
4. When that is complete, set the `OPTIMIZER_FEATURES_ENABLE` initialization parameter back to 11.2 and rerun SQL Performance Analyzer for the STS.

SQL Performance Analyzer produces a report that lists a SQL statement whose plan has regressed from 10g to 11g. For those SQL statements that are shown by SQL Performance Analyzer to incur performance regression due to the new optimizer version, you can capture their plans using an STS and then load them into the SMB.

This method represents the best form of the plan-seeding process because it helps prevent performance regressions while preserving performance improvements upon database upgrade.

Loading a SQL Plan Baseline Automatically



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Loading a SQL Plan Baseline Automatically: Scenario

Another upgrade scenario involves using the automatic SQL plan capture mechanism. In this case, set the `OPTIMIZER_FEATURES_ENABLE` initialization parameter to the pre-Oracle Database 11g version value for an initial period of time such as a quarter, and execute your workload after upgrade by using the automatic SQL plan capture.

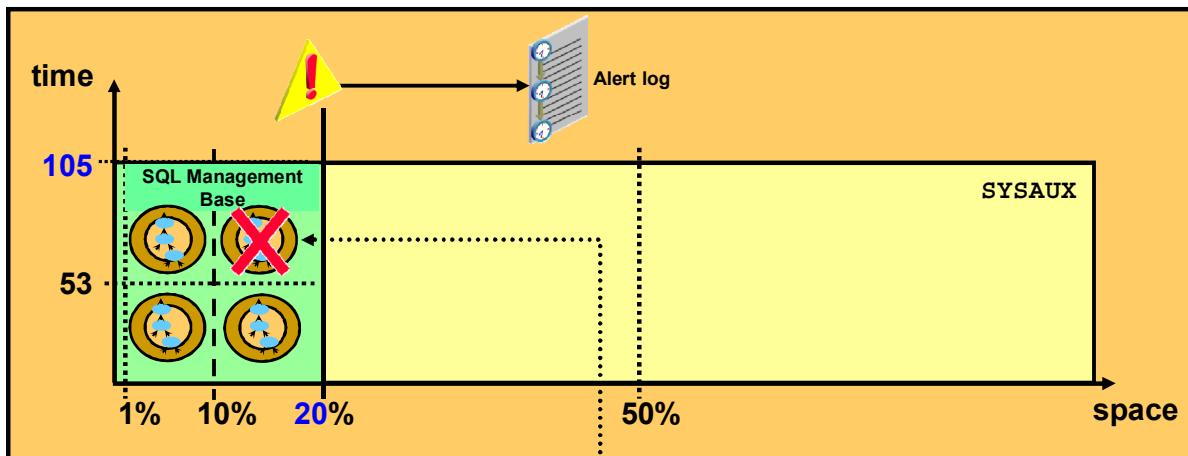
During this initial time period, because of the `OPTIMIZER_FEATURES_ENABLE` parameter setting, the optimizer is able to reproduce pre-Oracle Database 11g plans for a majority of the SQL statements. Because automatic SQL plan capture is also enabled during this period, the pre-Oracle Database 11g plans produced by the optimizer are captured as SQL plan baselines.

When the initial time period ends, you can remove the setting of `OPTIMIZER_FEATURES_ENABLE` to take advantage of the new optimizer version while incurring minimal or no plan regressions due to the plan baselines. Regressed plans will use the previous optimizer version; nonregressed statements will benefit from the new optimizer version.

Purging SQL Management Base Policy

```
SQL> exec dbms_spm.configure('SPACE_BUDGET_PERCENT',20);
SQL> exec dbms_spm.configure('PLAN_RETENTION_WEEKS',105);
```

DBA_SQL_MANAGEMENT_CONFIG



```
SQL> exec :cnt := dbms_spm.drop_sql_plan_baseline('SYS_SQL_37e0168b04e73ef');
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The space occupied by the SQL Management Base (SMB) is checked weekly against a defined limit. A limit based on the percentage size of the SYSAUX tablespace is defined. By default, the space budget limit for the SMB is set to 10 percent of SYSAUX size. However, you can configure SMB and change the space budget to a value between 1 percent and 50 percent by using the DBMS_SPM.CONFIGURE procedure.

If SMB space exceeds the defined percent limit, warnings are written to the alert log. Warnings are generated weekly until the SMB space limit is increased, the size of SYSAUX is increased, or the size of SMB is decreased by purging some of the SQL management objects (such as SQL plan baselines or SQL profiles).

The space management of SQL plan baselines is done proactively using a weekly purging task. The task runs as an automated task in the maintenance window. Any plan that has not been used for more than 53 weeks is purged. However, you can configure SMB and change the unused plan retention period to a value between 5 weeks and 523 weeks (a little more than 10 years). To do so, use the DBMS_SPM.CONFIGURE procedure.

You can look at the current configuration settings for the SMB by examining the DBA_SQL_MANAGEMENT_CONFIG view. In addition, you can manually purge the SMB by using the DBMS_SPM.DROP_SQL_PLAN_BASELINE function (as shown in the example in the slide).

Enterprise Manager and SQL Plan Baselines

Select Name	SQL Text	Enabled	Accepted	Fixed	Auto Purge	Created	Last Modified
<input type="checkbox"/> SQL_PLAN_buf0uhd3gc33z2d90e1d1	select signature, sql_handle, sql_text, plan_name...	YES	YES	NO	YES	Jul 31, 2009 10:50:41 PM	Jul 31, 2009
<input type="checkbox"/> SQL_PLAN_6zsnd8f6zsdf9g54bc8843	select /*LOAD AUTO*/ * from sh.sales where quantit...	YES	YES	NO	YES	Jul 31, 2009 10:50:18 PM	Jul 31, 2009
<input type="checkbox"/> SQL_PLAN_6zsnd8f6zsdf9g11df68d0	select /*LOAD AUTO*/ * from sh.sales where quantit...	YES	NO	NO	YES	Jul 31, 2009 10:51:52 PM	Jul 31, 2009

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Access the SQL Plan Control page by clicking SQL Plan Control in the Query Optimizer region of the Server page.

Use the SQL Plan Control page to manage SQL profiles, SQL patches, and SQL plan baselines from one location rather than separate locations in Enterprise Manager. You can also enable, disable, drop, pack, unpack, load, and evolve selected baselines.

From this page, you can also configure the various SQL plan baseline settings.

Using the MIGRATE_STORED_OUTLINE Functions

- Specify stored outlines to be migrated based on outline name, SQL text, or outline category, or migrate all stored outlines in the system to SQL plan baselines:

```
DBMS_SPM.MIGRATE_STORED_OUTLINE (
    attribute_name IN VARCHAR2,
    attribute_value IN CLOB,
    fixed IN VARCHAR2 := 'NO')
RETURN CLOB;
```

- Specify one or more stored outlines to be migrated:

```
DBMS_SPM.MIGRATE_STORED_OUTLINE (
    outln_list IN DBMS_SPM.NAME_LIST,
    fixed IN VARCHAR2 := 'NO')
RETURN CLOB;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use the DBMS_SPM.MIGRATE_STORED_OUTLINE function to migrate stored outlines for one or more SQL statements to plan baselines in the SQL management base (SMB). Specify which stored outlines to migrate based on the outline name, SQL text, or outline category. You can also migrate all stored outlines in the system to SQL plan baselines. Parameters are as follows:

- attribute_name**: Specifies the type of parameter used in **attribute_value** to identify the migrated stored outlines. Values (case-sensitive) are `outline_name`, `sql_text`, `category`, and `all`.
- attribute_value**: Based on the value specified in **attribute_name**. `NULL` if **attribute_name** is `all`.
- fixed**: Values of `NO` (default) and `YES`. Specifies the “fixed” status of the plans generated during migration. By default, plans are generated as “non-fixed” plans.

The second overload of the function is used to migrate stored outlines for one or more SQL statements to plan baselines in the SQL management base (SMB) given one or more outline names. Parameters are as follows:

- outln_list**: List of outline names to be migrated
- fixed**: Values of `NO` (default) and `YES`. Specifies the “fixed” status of the plans generated during migration. By default, plans are generated as “non-fixed” plans.

A CLOB is returned containing a formatted report describing the statistics during the migration.

Summary

In this lesson, you should have learned how to:

- Set up SQL Plan Management
- Set up various SQL Plan Management scenarios
- Migrate stored outlines to SQL Plan baselines



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Grid Control Architecture

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

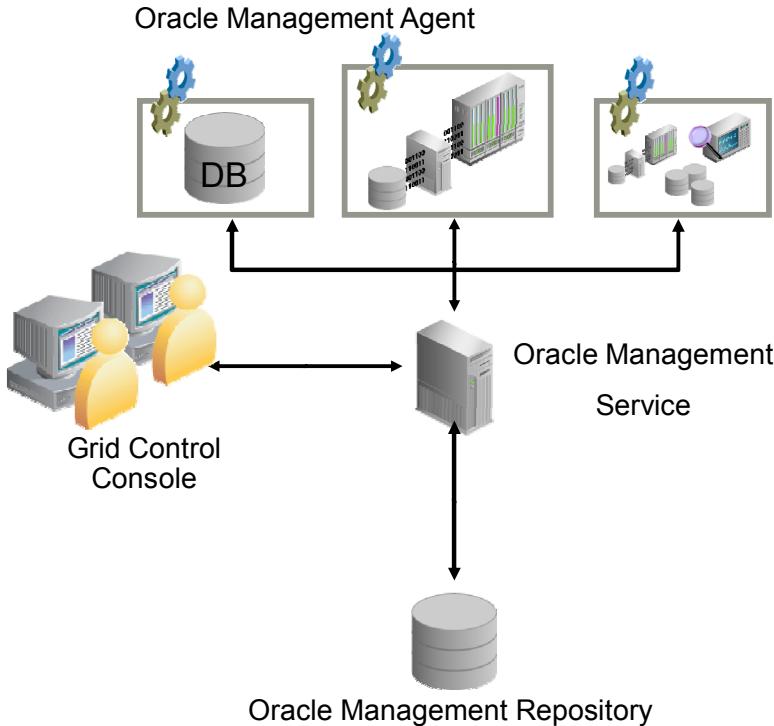
After completing this lesson, you should be able to:

- Describe the different components of Grid Control
- Explain the architecture of Grid Control
- List the target types managed by Grid Control
- Explain the Grid Control console pages and functionalities
- Discuss the application of the Maximum Availability Architecture to a Grid Control environment



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Grid Control Architecture



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Grid Control is composed of three main components:

- The Oracle Management Repository
- One or more Oracle Management Services
- One or more Oracle Management Agents

You can install Grid Control on a single server or on multiple servers, depending on the business needs and the size of the enterprise that is being managed.

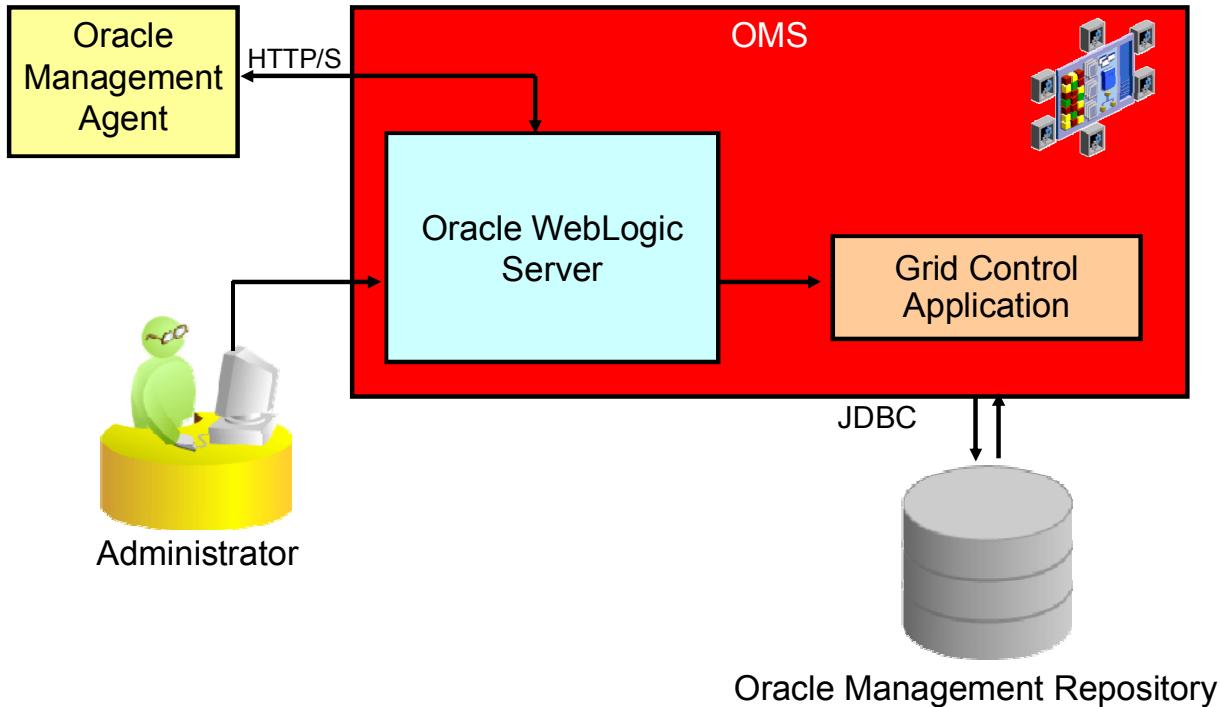
The Grid Control console is a Web-based user interface, which you use to manage and monitor targets in your data center.

The Oracle Management Agent (simply called an agent) is responsible for monitoring the health of the target. To monitor the target, you need to install the agent on the host on which the target runs. Regardless of the number of targets running, you need to install only one Oracle Management Agent per host—for example, to monitor Oracle Database and Oracle WebLogic Server running on a single host, you need only one agent. The agent collects information about target availability, configuration, and performance, and sends it through the Oracle Management Service (OMS) to the Oracle Management Repository. Each management agent talks to only one repository at any given time.

The OMS is a JEE Web application that renders the Grid Control console. It receives information from the agent or multiple agents and in turn saves it in the Oracle Management Repository. The OMS processes the monitoring and jobs information for each of the targets. Also, the OMS retrieves the data from the management repository and renders it into HTML for the browser to display.

The Oracle Management Repository contains a collection of Grid Control schema objects. It consists of objects such as database jobs, packages, procedures, views, and two tablespaces in an Oracle database that contain all available information about administrators, targets, and applications managed within Grid Control.

Oracle Management Service



ORACLE

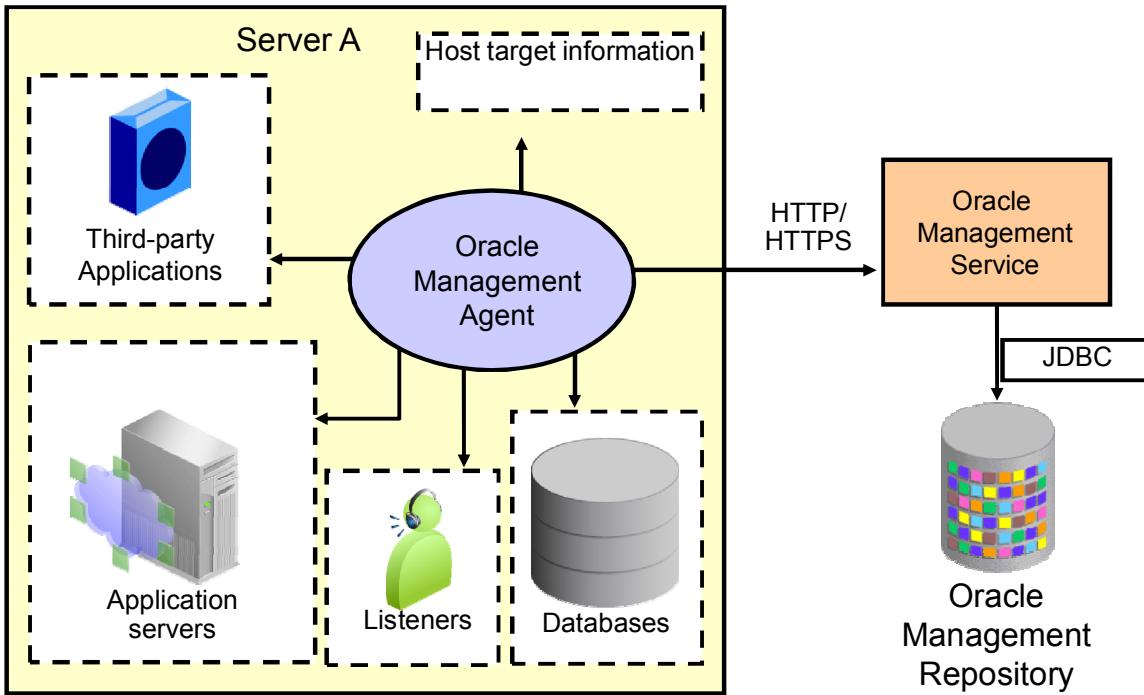
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The OMS renders the user interface for Grid Control, works with all management agents, and stores persistent data in Oracle Management Repository that it connects to using Java Database Connectivity (JDBC). It consists of two main components: Oracle WebLogic Server (serves HTTP requests) and the Grid Control application.

The Grid Control application produces the Grid Control user interface, which is delivered by OMS as HTML to a Web browser, using either Hypertext Transfer Protocol (HTTP) or Hypertext Transfer Protocol Secure (HTTPS).

You connect to the Grid Control application via a Web browser through Oracle WebLogic Server.

Oracle Management Agent



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Oracle Management Agent enables Grid Control to discover targets on a host and provides monitoring and administration capabilities for these targets. An agent is installed on each host server that is managed using Grid Control. As shown in the image in the slide, a single agent monitors all the targets on the host server, such as databases, database listeners, application servers, and also the host on which the agent is installed. The agent uploads all the configuration information of the targets to the OMS (middle tier). The agent is a system daemon that performs the following:

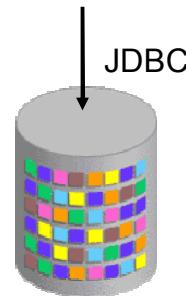
- Monitoring, alerting, and running jobs
- Ensuring that the agent is up and available

Agents are installed in their own ORACLE_HOME (unless you are using an NFS-mounted agent).

Oracle Management Repository

The Oracle Management Repository (OMR):

- Resides in an Oracle database
- Includes schema objects belonging to sysman
- Must be installed in a preexisting database
- Can be installed in a RAC database



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The OMR is installed in an Oracle database as a group of approximately 4,000 schema objects (stored in two tablespaces) belonging to the sysman user. These schema objects contain information about Grid Control administrators, targets, and applications that are managed by Grid Control. Oracle Net is used for communication between the OMS and the management repository via the database listener. By default, the listener monitors port 1521 for incoming connection requests. The OMR must be installed in a preexisting database, and for higher availability requirements can be installed in a Real Application Clusters (RAC) database.

Grid Control Targets

The commonly managed targets in Grid Control are:

- Oracle Database
- Fusion Middleware products
- Oracle Applications
- Third-party products



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Grid Control manages both Oracle and non-Oracle targets. Examples of a few commonly managed Oracle targets are:

- Oracle Database including 11g
- Oracle WebLogic Server
- Oracle SOA applications
- Oracle Identity Manager
- Oracle Siebel
- Oracle PeopleSoft

Grid Control Console: Home

The screenshot shows the Oracle Enterprise Manager 11g Grid Control Home page. At the top, there's a navigation bar with tabs: Home (highlighted with a red box), Targets, Deployments, Alerts, Compliance, Jobs, Reports, and My Oracle Support. The 'Home' tab is active. Below the navigation bar, the page is divided into several sections:

- Overview:** Total Monitored Targets: 6. A pie chart shows 80% Up (green) and 20% Unknown (grey).
- All Targets Status:** A legend indicates Unknown(1) and Up(4).
- All Targets Alerts:** Critical: 2, Warning: 0, Errors: 1.
- All Targets Policy Violations:** Critical: 25, Warning: 2, Informational: 0.
- All Targets Jobs:** Problem Executions (last 7 days): 0, Action Required Executions (last 7 days): 0, Suspended Executions (last 7 days): 0.
- Target Search:** Search bar with 'All' and a 'Go' button.
- Security Policy Violations:** Critical: 24, Warning: 2, Informational: 0. New in Last 24 Hours: 26.
- Recommended Security Patches:** Security Recommendations: 0. A note says 'Patch Recommendations information may be stale. My Oracle Support credentials are not configured.'
- Affected Targets:** 0.
- My Oracle Support Credentials:** Not Configured.
- Deployments Summary:** View: Database Installations. A table shows Database Installations: Oracle Database 11g 11.2.0.1.0, Targets: 1, Installations Applied: 1, Patches: No.
- Resource Center:** Links to Enterprise Manager Support Workbench and Oracle Technology Network.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Grid Control Console: Targets

The screenshot shows the Oracle Enterprise Manager 11g Grid Control interface. The top navigation bar includes links for Home, Targets, Deployments, Alerts, Compliance, Jobs, Reports, and My Oracle Support. The Targets link is highlighted with a red box, and the All Targets subtab within the Targets menu is also highlighted with a red box. Below the navigation, there is a search bar and a toolbar with options like Remove, Configure, Add, and Go. A table lists various targets with columns for Name, Status, and Type. The targets listed are:

Name	Status	Type
edrsr9p1.us.oracle.com	Green arrow	Host
edrsr9p1.us.oracle.com:3872	Green arrow	Agent
edrsr9p1.us.oracle.com_oms_csa_collector	Green arrow	CSA Collector
emrep.us.oracle.com	Blue circle with exclamation	Database Instance
LISTENER_edrsr9p1.us.oracle.com	Green arrow	Listener
Management Services and Repository	Green arrow	OMS and Repository

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Targets page enables you to view a list of the managed targets in your enterprise. At any time while using Grid Control, you can view a list of all the targets currently being managed by Grid Control, regardless of the target type. This can be useful if you want to scan a list of all the targets that you are responsible for managing in your environment. From the list of managed targets, you can quickly assess the availability of the targets and then drill down for more information.

To view a list of all the targets you manage, perform the following steps:

1. Click the **Targets** tab.
2. Click the **All Targets** subtab.

Grid Control displays a table listing all the targets that are currently being monitored.

Use the other subtabs on the Targets page to display only the host, database, middleware, Web application, service, system, and group or virtual server targets. In addition, you can customize the subtabs using the Preferences link at the top-right of the screen.

Grid Control Console: Deployments

The screenshot shows the Oracle Enterprise Manager 11g Grid Control interface. The top navigation bar includes Home, Targets, Deployments (which is highlighted with a red box), Alerts, Compliance, Jobs, Reports, and My Oracle Support. Below the navigation is a sub-menu with General, Provisioning, and Patches & Updates. The main content area has three main sections: 'Recommended Security Patches' (Security Recommendations: Unavailable, My Oracle Support: Credentials Not Set, with a note about patch recommendations being unavailable due to missing credentials), 'Deployments Summary' (View: Database Installations, showing a table for Oracle Database 11g 11.2.0.1.0 with 1 Target and 1 Patches Applied), and 'Configuration' (Search, Compare Configuration, Compare to Multiple Configurations(Job), View Saved Configurations, Download Policy Groups). To the right is an 'Overview' panel with the following text and bullet points:

Enterprise Manager maintains detailed information about hosts and their operating systems, as well as software installations. The Deployments Summary table provides a high level view of this information, as well as allowing you to explore the details by selecting the individual components. This information is also available on the Configuration page for a host. Using the Deployments tools, you can:

- Perform searches on the detailed information.
- Compare the detailed information of hosts and databases.
- Search My Oracle Support for patches, and subsequently manage their deployment.
- Clone a database or Oracle home to an alternative location.
- Manage the configuration collection process.
- Orchestrate Provisioning and Patching through Deployment Procedures.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Deployments page in Grid Control simplifies monitoring and management of your enterprise configuration. It serves as a starting point from where you can access many of Grid Control's configuration features, including provisioning and patching.

Grid Control Console: Alerts

The screenshot shows the Oracle Enterprise Manager 11g Grid Control interface. At the top, there's a navigation bar with links like Home, Targets, Deployments, Alerts (which is highlighted with a red box), Compliance, Jobs, Reports, and My Oracle Support. Below the navigation bar, there's a sub-navigation menu for 'Targets Down' with categories: Critical, Warning, Errors, Blacked Out, and Unknown Availability. A message says 'Your managed targets listed below are unavailable.' Below this is a 'Search' section with dropdowns for Target Type ('All Targets') and Alert triggered within ('31 Day(s)'), and a checkbox for 'Ignore acknowledged alerts'. A 'Go' button is next to the search fields. A table titled 'Targets Down' has columns: Target, Type, Alert Triggered, Message, and Acknowledged. The table body contains the message 'No Targets found..'. At the bottom of the page is a footer with links: Home, Targets, Deployments, Alerts, Compliance, Jobs, Reports, My Oracle Support, Setup, Preferences, Help, and Logout.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Grid Control provides a quick way to view all the alerts associated with a target. An alert indicates a potential problem, either a warning or critical threshold for a monitored metric has been crossed, or the target is no longer available. You can get critical alerts when a monitored metric has crossed its critical threshold or warning alerts when a monitored metric has crossed its warning threshold. Also, you can get alerts on various target availability states—for example, when the:

- Target is down
- Oracle Management Agent monitoring the target is unreachable
- Target is blacked out

Grid Control Console: Compliance

The screenshot shows the Oracle Enterprise Manager 11g Grid Control interface. The top navigation bar includes links for Home, Targets, Deployments, Alerts, Compliance (which is highlighted with a red box), Jobs, Reports, and My Oracle Support. Below the navigation is a sub-menu for Policies, Policy Groups, and Security At a Glance. The main content area is titled 'Policies: Violations' and features tabs for Violations, Library, Associations, and Errors. A search section titled 'Simple Search' allows filtering by Target Type (Host), Target Name, Category, and Severity, with options for 'Most Recent Violation within' and 'Ignore suppressed violations'. A table below shows a single row: '(No search conducted)'. At the bottom of the page is a footer with links to Home, Targets, Deployments, Alerts, Compliance (highlighted in blue), Jobs, Reports, My Oracle Support, Setup, Preferences, Help, and Logout.

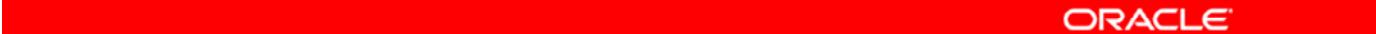
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use Grid Control to measure the compliance of a target with a standard. Compliance can be measured in a variety of ways. One way is to ensure conformance of a system's configuration to a standard. You can use policy groups to determine if a target complies with the standard. By evaluating a target against a policy group, you can determine whether a target complies with the guidelines of the standard, and if a target does not meet the desired state, what changes are required to make that target compliant.

Grid Control Console: Jobs

The screenshot shows the Oracle Enterprise Manager 11g Grid Control interface. The 'Jobs' tab is selected and highlighted with a red box. The main content area is titled 'Job Activity' and includes a search bar with 'Status: Active', 'Name' input, and 'Advanced Search' button. A note says 'TIP By default, results for the last 24 hours are displayed. Use 'Advanced Search' for more options.' Below the search is a 'View' dropdown set to 'Runs' and a 'Create Job' section with 'OS Command' selected. A table titled 'Select Name' lists no jobs found. The bottom of the page has a navigation bar with links like Home, Targets, Deployments, Alerts, Compliance, Jobs, Reports, My Oracle Support, Setup, Preferences, Help, and Logout.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Jobs tab displays all the jobs that are created in Grid Control. A job is a schedulable unit of work that you define to automate commonly run tasks. The OMS centrally controls jobs. At the time the job is scheduled to execute, information about the job is dispatched to the relevant management agents.

You can use jobs to accomplish a number of objectives, such as updating product release information and automating patch jobs. If you want a job to be available to other users for sharing and reuse, store the job in the job library.

Grid Control Console: Reports

The screenshot shows the Oracle Enterprise Manager 11g Grid Control interface. The top navigation bar includes links for Home, Targets, Deployments, Alerts, Compliance, Jobs, Reports (which is highlighted with a red box), and My Oracle Support. Below the navigation is a search section with fields for Title, Target Type (set to All), Owner (set to All), and Target Name, along with Go, Delete, Create Like, Edit, and Create buttons. A link to 'Expand All' or 'Collapse All' is also present. The main content area displays a table of report definitions:

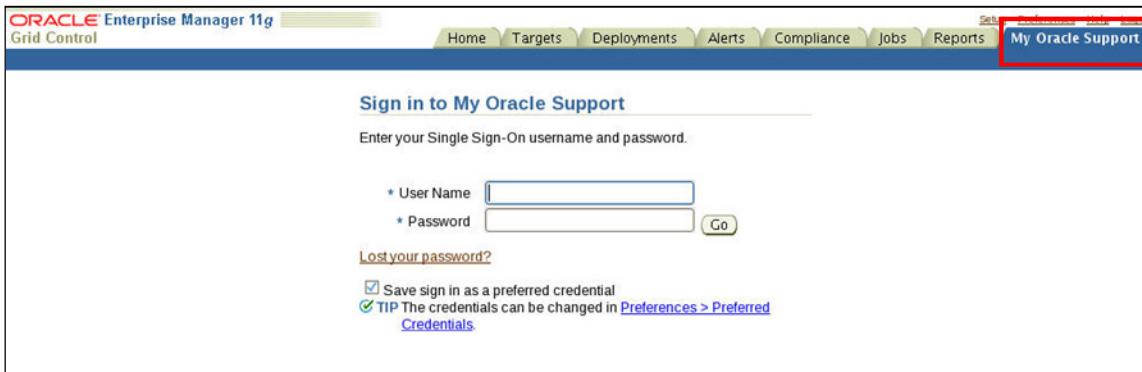
Select Title	Description	Date Generated	Owner
<input type="radio"/> Reports			
<input type="radio"/> Deployment and Configuration			
<input type="radio"/> Alerts and Policy Violations			
<input checked="" type="radio"/> IP Address Activity (Detailed) Report	Detailed report of IP Addresses with a high number of successes or failures, or a large number of distinct users. Report includes per user total.		SYSMAN
<input type="radio"/> IP Address Activity (Summary) Report	Report of IP Addresses with a high number of successes or failures, or a large number of distinct users.		SYSMAN

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Information Publisher provides ready-to-use report definitions, allowing you to generate reports immediately. You can also create custom report definitions to generate reports that satisfy unique information requirements.

The Report Definitions page displays all the report definitions to which you have access.

Grid Control Console: My Oracle Support



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Grid Control console also provides an integrated console to My Oracle Support, where you can get information about service requests, patches, and knowledge articles. Enter your My Oracle Support username and password, then click **Go** to move to the next screen.

Grid Control Console: My Oracle Support

The screenshot shows the Oracle Enterprise Manager Grid Control 11g interface. The top navigation bar includes links for Home, Targets, Deployments, Alerts, Compliance, Jobs, Reports, My Oracle Support, Setup, Preferences, Help, and Logout. The main content area is titled "Patch & Updates". It features three main sections: "Patching Quick Links" (Patch Recommendations), "Patch Related Activity" (Downloaded, Viewed, Reviewed, Favorites), and "Patch Search". The "Patch Search" section allows users to search by Patch ID or Number and Platform (Linux x86). Below these sections is a "Patch Plans" table:

Name	Status	Planned Deployment Date	Remove
10204gcplan4	Ready for Download	Not Specified	X
10204security	Ready for Download	Not Specified	X
111	Needs Validation	Not Specified	X
abcd	Needs Validation	Not Specified	X
abcd1	Needs Validation	Not Specified	X
db10.2.0.2plan	Needs Validation	Not Specified	X
db10.2.0.1plan3	Ready for Download	Not Specified	X

At the bottom of the page, there is a copyright notice: "Copyright (c) 2007, 2010, Oracle. All rights reserved. Legal Notices and Terms of Use | Privacy Statement | 3rd Party Licenses | About Oracle Enterprise Manager".

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

From My Oracle Support, you can look at patches and updates, the knowledge base, your service requests, and other information, as shown in the example in the slide.

Grid Control Console: Setup

The screenshot shows the Oracle Enterprise Manager 11g Grid Control Configuration interface. The top navigation bar includes links for Home, Targets, Deployments, Alerts, Compliance, Jobs, Reports, and My Oracle Support. The 'Setup' link is highlighted with a red box. The main content area is titled 'Overview of Setup' and contains sections for Enterprise Manager Configuration, Management Services and Repository, and Agents. A sidebar on the left lists various setup categories such as Roles, Administrators, Notification Methods, E-mail, Customization, Patching Setup, Blackouts, Registration, Passwords, Management Pack Access, Monitoring Templates, Corrective Action Library, Management Plug-ins, and Management Connectors.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Click the Setup link at top of the Grid Control console to set up your Grid Control environment. You can use the Setup page to create administrators other than the default sysman administrator account for daily administration work. Oracle recommends that these administrators reflect your management team. Assign the administrators roles and privileges to segregate their responsibilities. Oracle also recommends that no one shares an account.

Grid Control Console: Preferences

The screenshot shows the Oracle Enterprise Manager 11g Grid Control Preferences page. The left sidebar has a 'General' tab selected. The main content area has two sections: 'Password' and 'E-mail Addresses'. The 'Password' section contains a note about changing the SYSMAN user password. The 'E-mail Addresses' section shows a table with one row: 'No e-mail addresses specified'. There are 'Revert' and 'Apply' buttons at the bottom of each section. At the very bottom, there is a navigation bar with links like Home, Targets, Deployments, etc., followed by 'Setup', 'Preferences', 'Help', and 'Logout'. A red box highlights the 'Logout' link.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Preferences page enables you to set the preferred credentials for the targets monitored by Grid Control. You can also use this page to specify one or more email addresses. In the event Grid Control generates an alert that triggers an email notification, an email is sent to the email address that you specify on the Preference page. In addition, you can customize the subtabs by using Target Subtabs.

Grid Control High Availability

- Service disruption can result from failures in any of the Grid Control components:
 - Management agent failures
 - OMS failures
 - Repository failures
- Grid Control can be deployed in a variety of configurations that provide greater or lesser degrees of protection.
 - Backup and recovery mode
 - Active/Passive mode
 - Active/Active mode
 - Disaster recovery mode



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The impacts of failure of the different Grid Control components are:

- **Management agent failure or failure in the communication between management agents and OMS:** This results in targets no longer being monitored by Grid Control, though the Grid Control console is still available and you can view historical data from the repository.
- **Oracle Management Service (OMS) failure:** This results in the unavailability of the Grid Control console as well as unavailability of almost all Grid Control services.
- **Repository failure:** This results in failure on the part of Grid Control to save the data uploaded by the agents. This also results in the unavailability of almost all Grid Control services.

You can prevent these failures by deploying Grid Control in a highly available manner, as recommended under Oracle's Maximum Availability Architecture (MAA). Grid Control can be configured in:

- **Backup and recovery mode**, using standard database tools for repository backup/recovery
- **Active/Passive mode**, involving two hardware nodes with a single node running an OMS instance (active) at a time
- **Active/Active mode**, where two or more OMS instances are active at the same time
- **Disaster recovery mode**, using physical standby database technology to provide failover for the repository from one machine to another

Quiz

Identify the Grid Control components.

- a. Oracle Management Agent
- b. Oracle Management Service
- c. Oracle Management Repository
- d. All of the above



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Answer: d

Summary

In this lesson, you should have learned how to:

- Describe the different components of Grid Control
- Explain the architecture of Grid Control
- List the target types managed by Grid Control
- Explain the Grid Control console pages and functionalities
- Discuss the application of the MAA to a Grid Control environment



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

8

Grid Control Installation

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Install Grid Control
- Describe the installation process for the various components that make up Grid Control
- Discuss the ports used for Grid Control installation
- Explain the directory structure of Grid Control



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Installing Grid Control

The installation process for Grid Control 11g can be divided into three major sections:

- Oracle Database installation
- Oracle WebLogic Server installation
- Grid Control installation



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The installation process for Grid Control has three main sections. Because it requires a preexisting certified database for the Oracle Management Repository (OMR), you must first have access to an existing certified Oracle Database installation, or you should install a certified Oracle Database, which will host the Grid Control repository. Database releases that are certified for use as an OMR are documented in Note 412431.1, available on My Oracle Support. In this course, you use an Oracle Database 11g installation for the repository. For further information about installing Oracle Database 11g, see the database installation guide in the Oracle Database Documentation Library.

Likewise, the Grid Control installation requires an Oracle WebLogic Server running on the same host where you will be installing Grid Control. Because the WebLogic Server installation is a new requirement for the installation of this release of Grid Control, the next few slides take you through a typical Oracle WebLogic Server installation.

Oracle WebLogic Server Installation

Oracle WebLogic Server can be installed in one of three modes:

- Console mode
- Silent mode
- GUI mode

Only GUI mode installation is documented in this course.

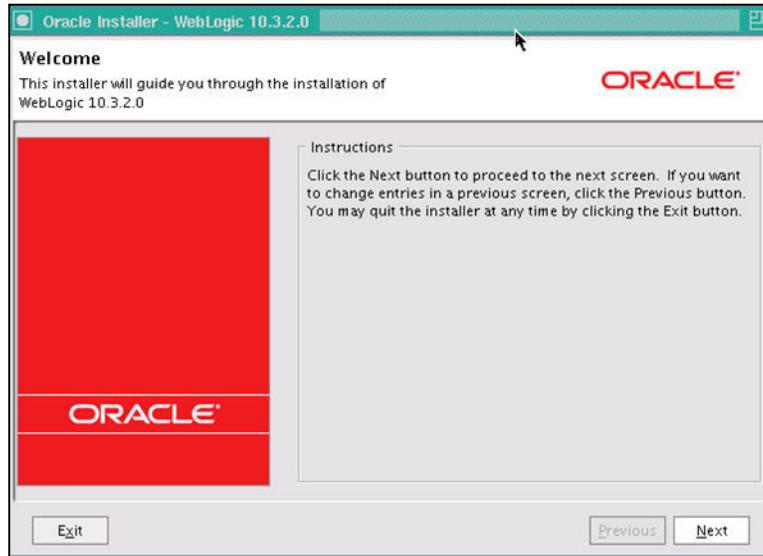


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle WebLogic Server can be installed in three modes—console, silent, and GUI. Console and silent mode installations are beyond the scope of this course. GUI mode installation is the graphics-based method of executing the Oracle WebLogic Server installation program. It can be run on both Windows and UNIX systems.

The installer involves initially downloading a small piece of software, selecting the installation options, and then downloading and installing only the components that you select. The Net installer eliminates the need to download a large, single executable binaries file before actually installing the product.

Oracle Installer: Welcome



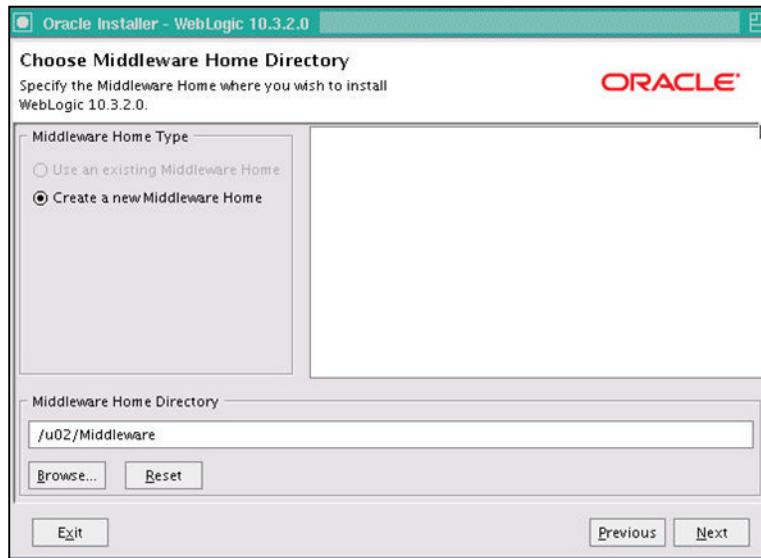
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

The WebLogic Server installation process starts with a Welcome screen. Simply click the **Next** button to progress to the next screen.

You may cancel the installation at any time by clicking **Exit**.

Choose Middleware Home Directory



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

The next screen prompts you for a Middleware home directory. If you are installing to a machine where Middleware is already installed, you may use the existing Middleware home (recommended) or create a new Middleware home. If you create a new Middleware home, the Oracle WebLogic Server installer program creates the directory for you. Click the **Next** button to progress to the next screen.

Register for Security Updates

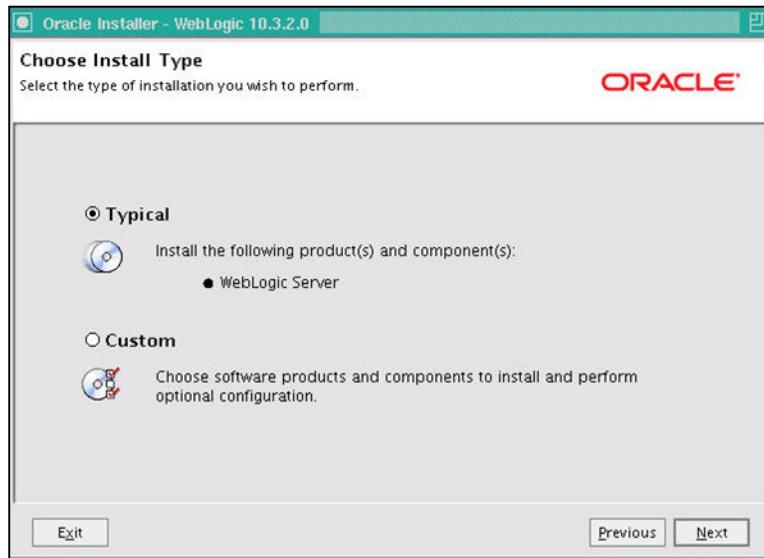


ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The next screen allows you to provide your email address to be informed about security issues. You can also receive security updates via My Oracle Support. Click **Next** to move to the next screen.

Choose Install Type

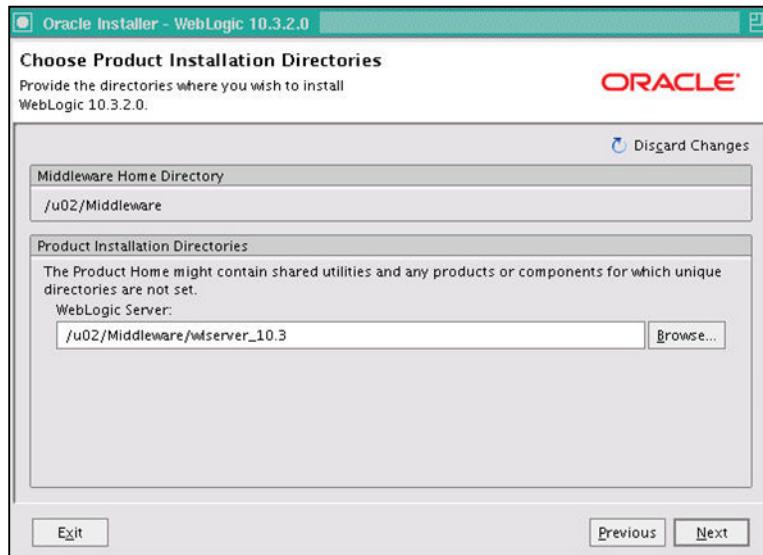


ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The next screen prompts you to select the type of installation you want. A typical Oracle WebLogic Server 10.3 installation includes both the server components and the examples and workshop for the Oracle WebLogic Server platform. In this release of Grid Control, only the Typical installation is supported, so select Typical, and click the **Next** button. The Custom installation is not documented further here.

Choose Product Installation Directories

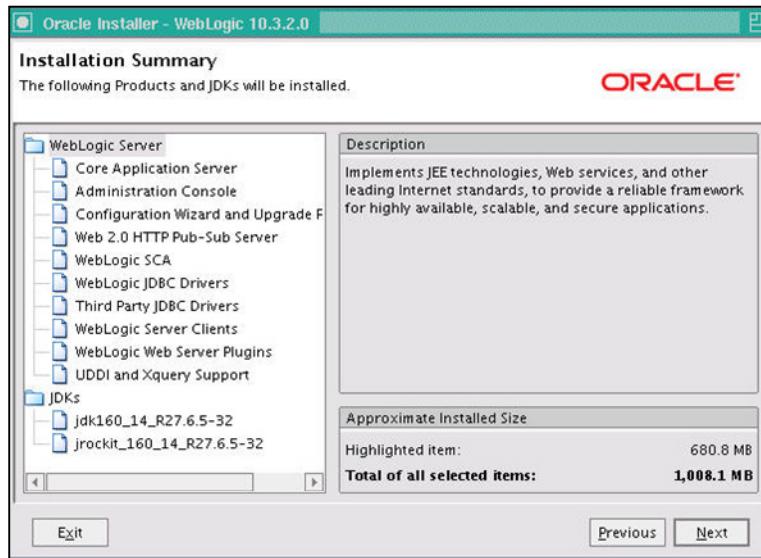


ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The next screen allows you to specify a product installation directory name under the Middleware home directory you specified on an earlier screen. Enter a Product Home (or accept the default), and click the **Next** button to progress to the next screen.

Installation Summary

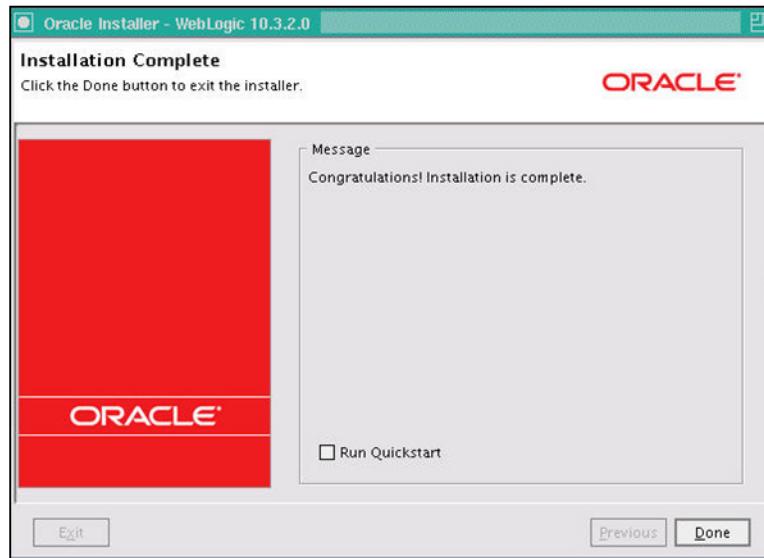


ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Installation Summary screen summarizes the products and Java Development Kits (JDks) that will be installed. Click the **Next** button to move to the next screen.

Installation Complete



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When the product installation is completed, the Quickstart application is selected by default. Because Quickstart is not a requirement for Grid Control, you can deselect the Run Quickstart check box on the Installation Complete page. Click **Done** to finish the installation process.

Installing Grid Control

After the Oracle Database and WebLogic Server software is installed and configured, and the repository database is built, you can proceed to the installation and configuration of Grid Control itself. Grid Control software is available on:

- DVD-ROM
- The OTN Web site



You can install Grid Control using:

- The Oracle Universal Installer (OUI)
- The response file for silent installation

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Grid Control software is available on a DVD-ROM or you can download it from the Oracle Technology Network (OTN) Web site. You can access and install Enterprise Manager Grid Control by:

- **Using a Remote DVD Drive (UNIX only):** If the computer on which you want to install Grid Control does not have a DVD drive, you can perform the installation from a remote DVD drive by mounting (sharing) that DVD drive.
- Staging the DVD and installing from the staged software
- **Using Remote Access Software for Remote Computers:** If the computer on which you want to install Grid Control is on a remote location and you do not have physical access to the computer, you can perform the installation provided you can access the computer using a remote access software.

You can install Grid Control using the OUI. The OUI is a Java-based graphical user interface application that enables you to install Oracle components from a DVD, multiple DVDs, or the Web. Grid Control supports silent installations as well those in which you can install the Grid Control components (management service, additional management service, or management agent) without going through an interview phase (displaying pages or responding to questions). In silent installations, a response file provides the necessary installation information, typically answered by the user, using stored values.

My Oracle Support Details



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The first screen of the Grid Control installation allows you to provide email or My Oracle Support details so that you can be informed about security issues and updates. You can also click the Installed Products button to see a list of products installed on this machine (this will query the Oracle Inventory to display products and version numbers), or at any stage you can click the Cancel button to cancel the installation without updating the inventory. Note that providing your details on this screen is optional; you can leave the Email field blank and deselect the check box to receive security updates, and still install Grid Control successfully. If you do not provide your details, you will be warned that you may remain uninformed about important security updates.

Click **Next** to progress to the next screen.

Check for Updates



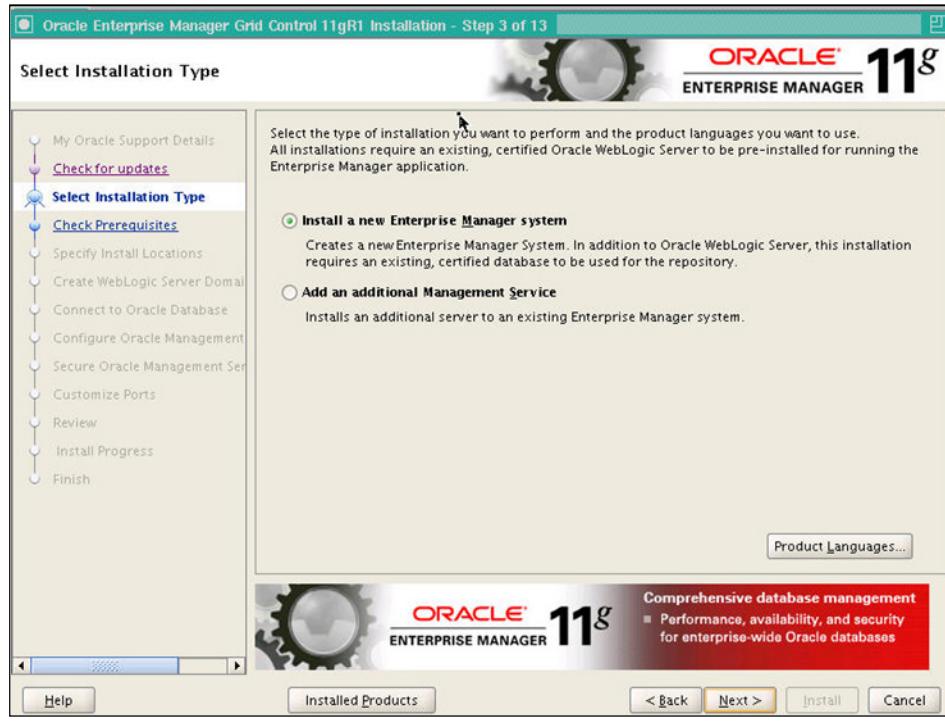
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On this screen, you can specify the location where software updates can be automatically installed while the installation is progressing. Software updates may include one-off patches, critical updates, prerequisite updates, install updates, patch set releases, and so on. These updates will ensure that the Grid Control installation is successful. You can:

- Download and install updates from My Oracle Support. Select this option to allow the installer to connect to My Oracle Support and automatically download the relevant updates. Enter your My Oracle Support username and password (and proxy information if relevant). You can test whether the information is correct by clicking the **Test Connection** button.
- Install updates from a staging location. Select this option if you have already downloaded software updates locally. Enter the location directly or click the **Browse** button to search for the relevant location.
- Skip software updates. Select this option if you do not want to search for software updates, or if you are certain there are no relevant updates available.

Click **Next** to progress to the next screen.

Select Installation Type



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

This screen allows you to select the type of installation you want to perform, as well as the product language you want to use. The two options that are available to you are:

- **Install a new Enterprise Manager system:** Select this option to install Grid Control using an existing certified Oracle database. By default, this option is selected when you invoke the Installer. This option will install the following components:
 - Oracle Management Service 11g
 - Oracle Management Agent 11g

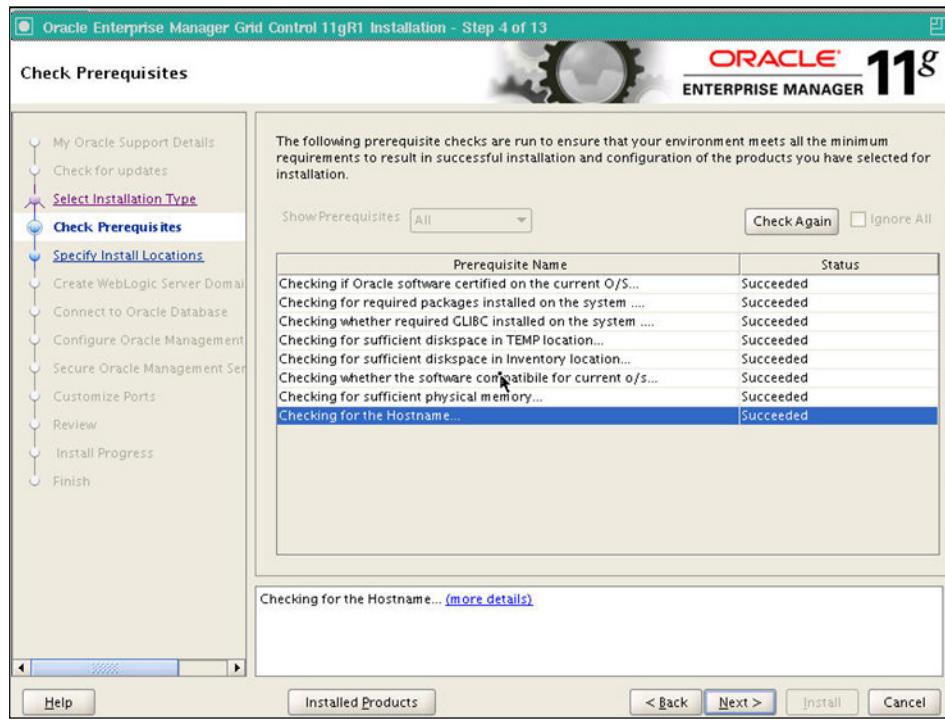
It also configures the Management Repository in the existing, certified Oracle database.

- **Add an additional Management Service:** Select this option to install an additional Oracle Management Service. You must already have an existing Grid Control installation to use this option.

Click **Next** to accept the default and progress to the next screen.

Note: If this is the first installation of an Oracle product you have performed on this machine, a screen will appear when you click Next asking for an inventory location. Otherwise this screen is skipped.

Check Prerequisites



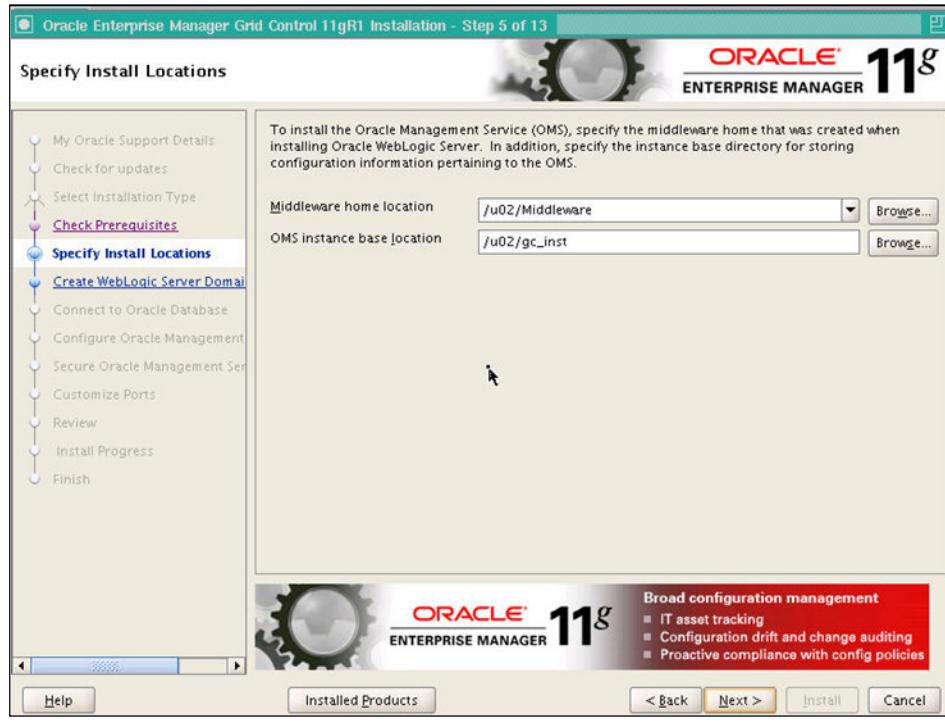
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Check Prerequisites screen is displayed. At this point, the Installer runs some prerequisite checks to verify if the environment meets the minimum requirements for a successful Grid Control installation. For further information about the minimum requirements, see the relevant appendices of the *Oracle Enterprise Manager Grid Control Basic Installation Guide 11g Release 1 (11.1.0.1.0)-part number E15838-01*.

When the checks are completed, click the **Next** button to progress to the next screen.

Specify Install Locations

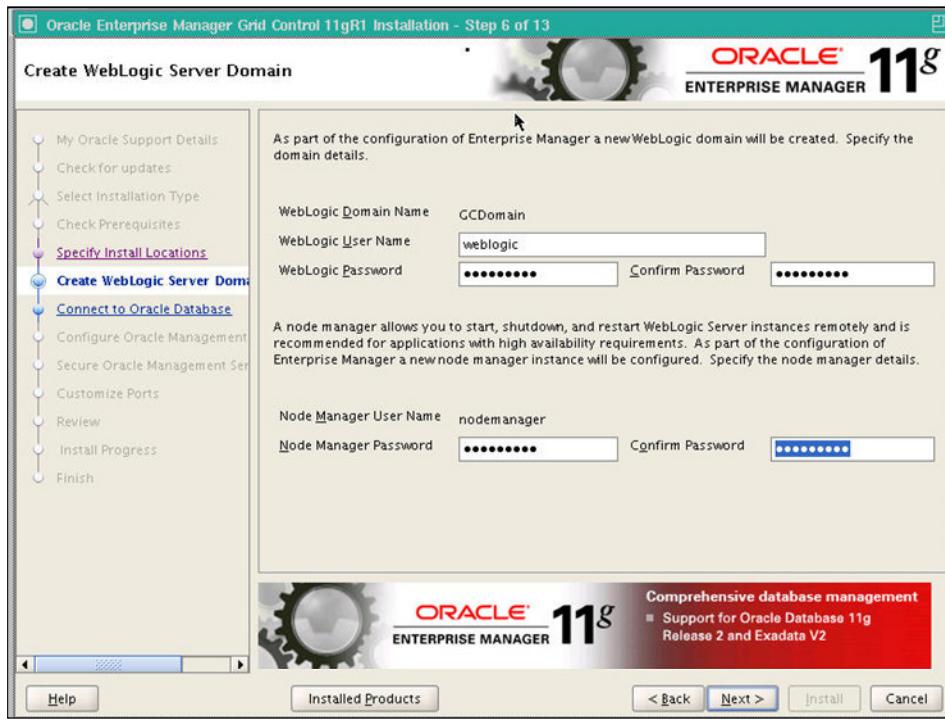


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

On the Specify Installation Locations screen, you specify paths to the Oracle Middleware home (the OMS, WebLogic Server, and agent software are installed under this directory, for example, under /u02/Middleware in the example in the slide) and the Oracle Management Service instance base location (which contains configuration and log files for the OMS). Accept the defaults or enter modified values, and click **Next** to move to the next screen.

Create WebLogic Server Domain



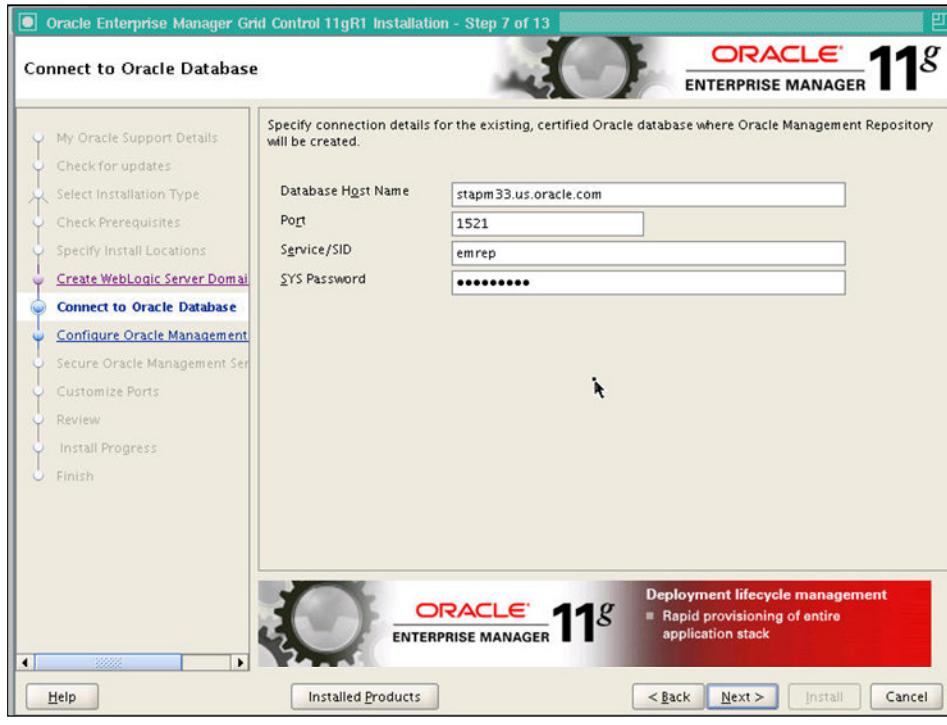
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

The Create WebLogic Server Domain screen prompts you for a WebLogic username and password, as well as a node manager username and password. You can accept the default domain name and usernames, but Oracle recommends you enter sufficiently strong passwords that include a mixture of upper and lowercase letters, as well as numbers (for example, XwltyRsp97).

By default, the WebLogic domain name is GCDomain (the installation creates a new domain with this name), and the Node Manager username is nodemanager. These are noneditable fields. The installer uses this information for creating the Oracle WebLogic Domain and other associated components, such as the admin server, the managed server, and the node manager. Click the **Next** button to progress to the next screen.

Connect to Oracle Database



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

This screen prompts you for connection information for the Oracle database where the Oracle Management Repository (`sysman` schema) will be created.

If you used the GUI installation process for the Oracle database, you will need to deconfigure Database Control before this database can be used for the Oracle Management Repository. To do that, enter commands similar to the following from a terminal session (**Note:** These commands are used with Bash shell. You will need to modify them suitably for other shells or Windows environments. You should also log on as the Oracle software owner to run these commands):

```
$ export ORACLE_HOME=<full_path_to_Oracle_database_software_location>
$ export PATH=$ORACLE_HOME/bin:$PATH
$ emca -deconfig dbcontrol db -repos drop -SYS_PWD <SYS_Password> -
SYSMAN_PWD <SYSMAN_Password>
```

When prompted, enter a value for the Database SID and Listener port number, and press **Y** to continue.

When the command completes, return to the Grid Control installation process and enter a database host name, port, service/SID and SYS password, then click **Next**.

Configure Oracle Management Repository



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On the next screen, you are prompted for passwords for the sysman user. sysman is the Super Administrator for the Oracle Management Repository. Again, ensure that you enter sufficiently strong passwords that include a mixture of upper and lowercase letters, as well as numbers (for example, XwltyRsp97), and that the passwords are different from each other.

You also need to provide data file locations for the management repository schema objects (for more information about these data files, refer to the basic installation guide available at <http://otn.oracle.com/documentation/oem.html>). Accept the default data file locations or enter new values, and then click the **Next** button to proceed to the next screen.

Secure Oracle Management Service



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

On this screen you configure security for the Oracle Management Service. Specify a registration password that will be used to secure the communication between the OMS and Oracle Management Agents. The password you provide must be at least eight characters long, start with a letter, and include at least one numeric value, one uppercase letter, and one lowercase letter.

For more enhanced security, you can configure the OMS in such a way that only secured agents can communicate with it. You can also configure the Grid Control console so that it can be accessed only by HTTPS (the default is to allow either HTTP or HTTPS protocol access).

Click the **Next** button to progress to the next screen.

Customize Ports



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Customize Ports screen allows you to modify the default ports Grid Control uses for specific components. These defaults come from the `staticports.ini` file if the installer is invoked with that file. Otherwise, the first free port from the recommended port range as listed is used. You may override these defaults if, for example, you already use some of these ports for other software products. If you decide to override the defaults, ensure that you select a port that is not already used.

Click the **Next** button to progress to the next screen.

Review

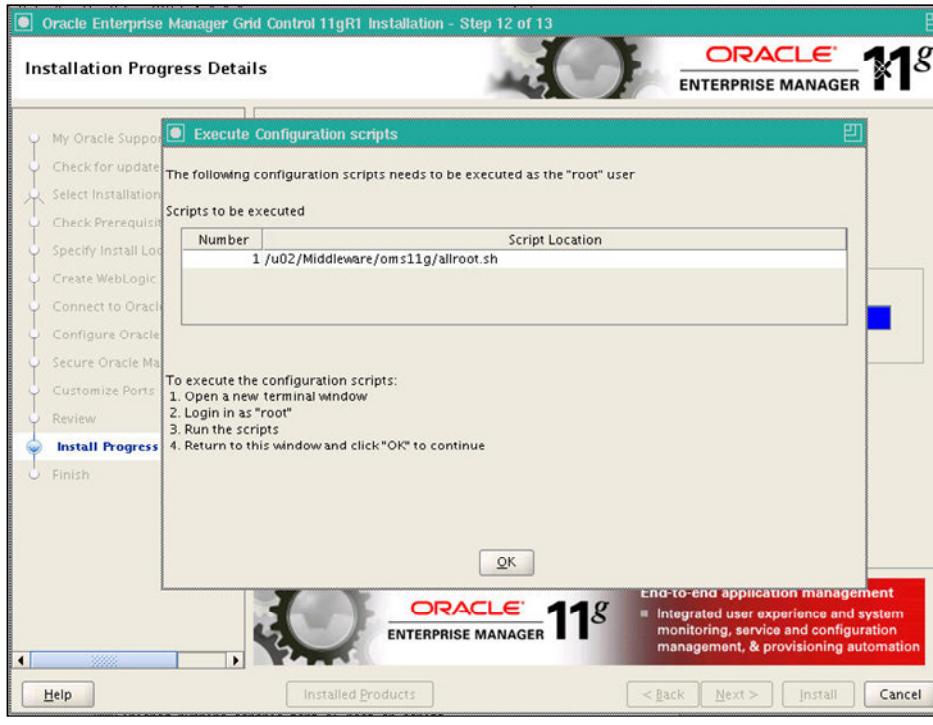


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Review screen enables you to review the information you have provided so far during the interview process. You can expand any of the information provided if it has not already been expanded by clicking the plus sign to the left of the area you are interested in expanding (for example, the Ports section in the screenshot in the slide).

If you want to change any of the information listed, click the Back button to go back to the screen you need to change the information on. Otherwise, click the **Install** button to begin the installation process.

Execute Configuration Scripts

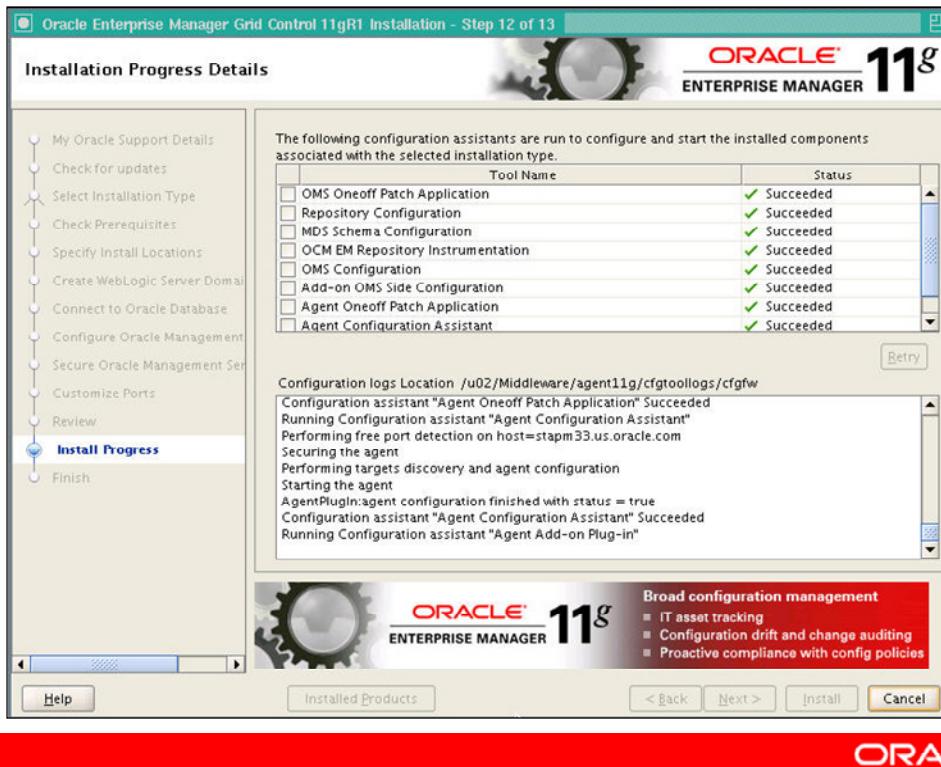


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The installation now proceeds. Part way through the installation, you are prompted to run a configuration script. This needs to be run as the “root” user, so either log on as that user and run the script or ask your system administrator to run the script for you.

After the configuration script has completed, return to this window and click the **OK** button to progress to the next screen.

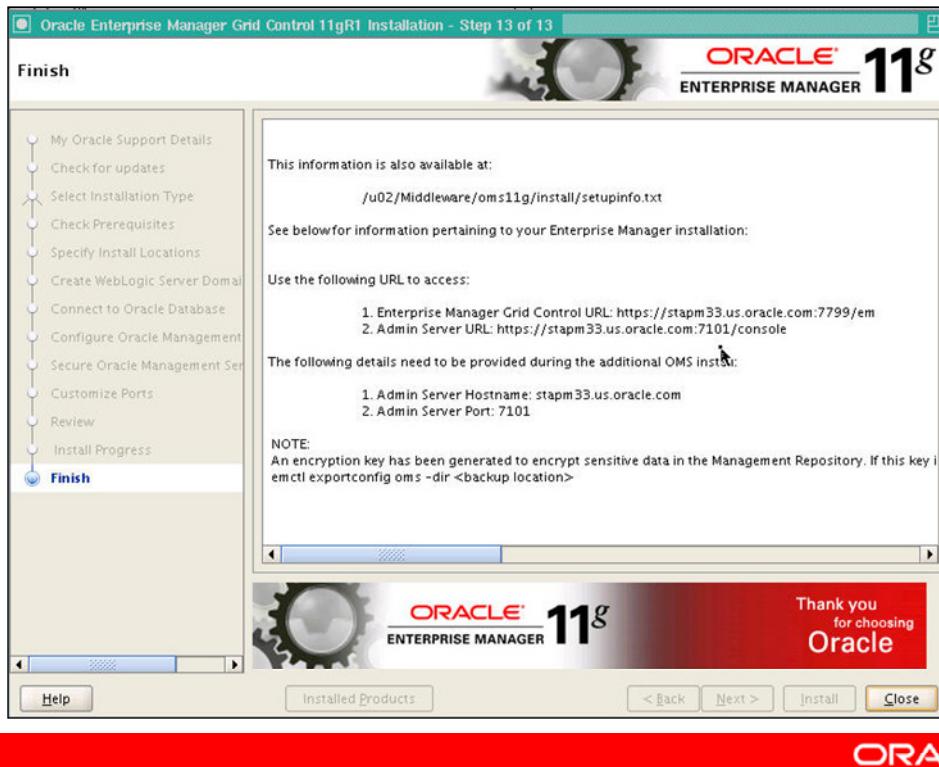
Installation Progress Details



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On this screen, a number of configuration assistants run, which configure and start the components you have installed. If any of the configuration assistants fail, you can select it and click the Retry button to retry that particular configuration assistant. The log file locations are also displayed so you can investigate the problem further before retrying.

Finish



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Finish screen appears when the installation is complete. It also shows you the URLs to access both the Grid Control console and the Admin Server. This information is also available in the Oracle Management Server installation under <OMS_HOME>/install/setupinfo.txt.

Click the **Close** button to exit the installer.

Oracle Management Agent Installation

The Oracle Management Agent can be installed in multiple ways.

- Installation media
- Mass deployment:
 - Push
 - Pull
 - Network File System (NFS)
 - Clone
 - RPM Package Manager (RPM)



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Oracle Management Agent can be installed in a variety of ways. For less complex environments, installation can be done simply by loading the installation media on the box the agent is being installed on, and running the installer silently.

However, many customers have hundreds or even thousands of targets to monitor, and using that approach in such environments would be incredibly tedious. For these customers, mass deployment options reduce the time and resources needed to deploy agents to many hosts. These mass deployment methods also support multiple communication methods and security models. The options are:

- Using the Agent Deploy application to specify a list of hosts and “push” agents onto those machines. This feature uses the Secure Shell (SSH) protocol to connect to the destination hosts and copy the required installation files and scripts.
- Running the `agentDownload.<platform>` script, which invokes the `wget` utility to “pull” the Oracle Universal Installer and installation response file from the OMS host to the agent host. The installer is then executed in silent (noninteractive) mode, using the values specified in the response file to install the agent.
- Installing the agent on a shared NFS mount point, then running the `nfsagentinstall` script from each host the agent should be installed on

Note: NFS mounted agent deployment is not supported on clusters.

- Cloning an existing agent installation that is known to be working, patched to the appropriate level and certified for use. This functionality is available by executing command-line utilities and from the Grid control Console starting from version 10.2.0.5.0 or later.
- Downloading the agent RPM kit from http://www.oracle.com/technology/software/products/oem/htdocs/provisioning_agent.html, then installing the rpm as root using `rpm -ivh <absolute location for agent rpm>`

OMA Installation: Agent Push

You start Agent Push by:

1. Configuring SSH between the OMS host and the target hosts
2. Clicking the **Deployments** tab
3. Clicking **Install Agent** on the General subtab
4. Clicking **Fresh Install**



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Agent Push installation method is one of the most straightforward ways to install an Oracle Management Agent onto one or more target machines. The connectivity between OMS and the specified hosts is established using the SSH protocol. In Grid Control 11g, the Agent Deploy application sets up and drops SSH connectivity automatically for Linux/UNIX environments. However, Windows hosts do not usually support SSH access by default. If the management agent is going to be on a Microsoft Windows operating system, you need to manually install and configure the Cygwin suite on that host to allow the OMS to connect to it. The agent deployment application needs SSH services along with other software tools that come with the Cygwin suite. The full suite can be downloaded from <http://www.cygwin.com>.

After SSH is successfully set up and tested between the OMS host and the target hosts, click the **Deployments** tab, and then click **Install Agent** on the General subtab. Click **Fresh Install** to start the interview process.

OMA Installation: Agent Push

The screenshot shows the 'New Agent Installation: Installation Details' screen in Oracle Enterprise Manager 11g Grid Control. The 'Source Software' section has 'Default from Management Server location' selected. The 'Hosts' section shows a host list 'srbd09'. The 'OS Credential' section shows 'emha' as the username and '*****' as the password. The 'Management Server Security' section shows a registration password '*****'.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The New Agent Installation: Installation Details screen presents a sequence of installation options.

The recommended way of downloading the agent software from OTN is to use the agent download feature of Grid Control. This feature can download the agent software to the default location or to another location (refer to the Advanced Install guide available at <http://otn.oracle.com/documentation/oem.html> for more details). The best practice for the source shiphomedirectory is to use the default option, which is the location within the management server. It is worth noting that this location is the Agent Download directory of the OMS (for example, <OMS_HOME>/sysman/agent_download/<version>/<platform>). If you are not using the default location, make sure that the shiphomedirectory you provide is shared and visible by all the hosts on which the agent has to be installed.

The hosts section is used to define the hosts that are to have agents installed on them, as well as their platform. Only one platform type can be deployed to in a single operation; in order to deploy agents onto Linux, Windows, and Solaris hosts, for example, a user would be required to perform three operations (one for each set of machines on a single platform). Also, the list of platforms available in the drop-down menu corresponds to the platforms for which installation software exists in the Agent Download directory of the OMS.

The operating system credentials entered on the details screen must be the same for all hosts being deployed to in the current deployment operation. If you are performing the installation as a user with “sudo” privileges, you can use these privileges to automatically run the `root.sh` script as well.

The `<ORACLE_HOME>` destination for the agent installation will be the same for all hosts being deployed to in the current deployment operation. Ensure that the specified base installation directory is empty. If you leave the port empty, the application will pick up the next free port on its own. If you provide a port that is already being used, you will get a warning during the prerequisite checks and you can change the port then or force the application to use the same port.

The Management Server Registration Password can be optionally provided in the Management Server Security section further down the page. Providing the password will configure the agents to run in secure mode after installation. In Grid Control, the communication between the Oracle Management Service (OMS) and the agent can either be secure (over HTTPS between a secured agent and the OMS) or insecure (over HTTP between an insecure agent and the OMS). The OMS runs in a secure mode and can further have a “locked” or “unlocked” state. If it is unlocked, both secure and insecure agents can talk to the OMS. If it is locked, only secure agents can talk to it. The password should be provided depending on the requirement of secure communication and the OMS state. You can also secure the agent later sometime by issuing the following command:

```
<AGENT_HOME>/bin/emctl secure agent
```

Finally, the application offers the ability to run additional scripts both before and after installation. If you are performing the installation as a user with “sudo” privileges, you can use the `sudo` privileges to run these additional scripts as `root`.

OMA Installation: Agent Push

The top screenshot shows the 'My Oracle Support Details' configuration page. It includes fields for My Oracle Support Email Address, a checkbox for receiving security updates, and connection details for a proxy server (Proxy Server, Proxy Port, Proxy Username, Proxy Password). The bottom screenshot shows the Oracle Enterprise Manager 11g Grid Control interface during the 'Installing' phase of the agent push. It displays a progress bar and a list of prerequisite checks that have been successfully completed.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



This screen prompts you to provide My Oracle Support details, as well as connection details for the proxy server for the host target (if needed). Click the **Submit** button to start the installation process. Next, connectivity between the OMS host and target hosts is checked, followed by a series of prerequisite checks executed on the remote hosts. If any failures in connectivity occur, or prerequisites fail or result in warnings, the application displays a page explaining the nature of the failures or warnings, and how they can be corrected. (For nonfatal warnings, the user is offered the opportunity to ignore the messages and continue.) If no problems were found during the prerequisite checks, the actual installation begins.

OMA Installation: Agent Push

Session Information

Session Initiated Date Time: Thursday, January 7, 2010 3:57:12 PM PST

User Inputs

Install Type	New Install
Version	11.1.0.10
Source Shippoint Directory	Default, from Management Server location.
Installation Base	/139.185.35.109
Remote Host Names	139.185.35.109
Allow local hostname to override hostnames	Not Selected
Run nodt.sh	Not Selected
Load Balancer Host	Not Specified
Load Balancer Port	Not Specified
Additional Parameters	Not Specified
Management Server Registration	Not Specified
Pre-Install Script	Not Specified
Post-Install Script	Not Specified

Prerequisite Details

To view the prerequisite details [Click Here](#)

Host Name	Status	Application Prerequisite Logs	System Prerequisite Logs
localnode	Success	<ORACLE_HOME>/sysman/prov/agentpush/2010-01-07_15-57-12-PM/prereqs/entrypoints/connectivity/focal	Not Applicable
139.185.35.109	Success	<ORACLE_HOME>/sysman/prov/agentpush/2010-01-07_15-57-12-PM/prereqs/entrypoints/eragent_install/139.185.35.109	<ORACLE_HOME>/sysman/prov/agentpush/2010-01-07_15-57-12-PM/prereqs/entrypoints/oracle.sysman.top.agent_Complete/139.185.35.109

Install Details

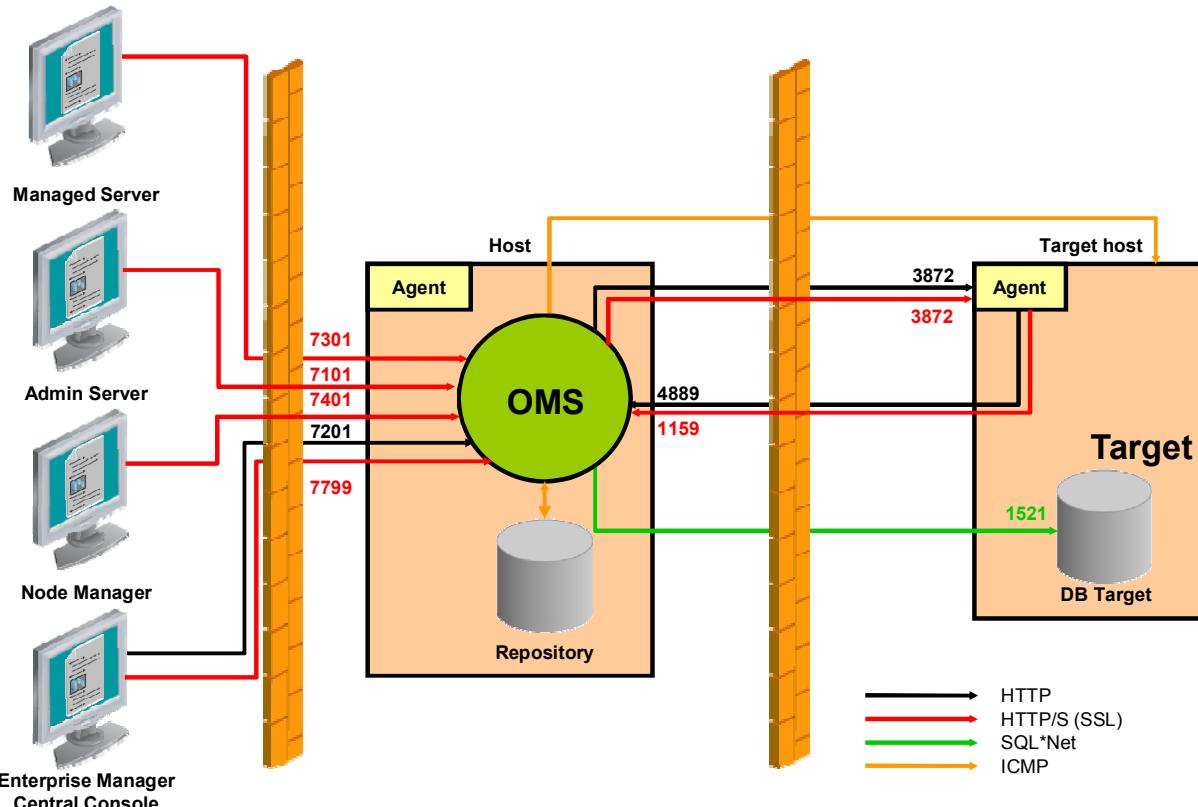
The Installation on all the nodes was Successful

Host Name	Status	Agent Status	Logs
139.185.35.109	Success	<ORACLE_HOME>/sysman/prov/agentpush/2010-01-07_15-57-12-PM/logs/139.185.35.109/agentStatus.log	<ORACLE_HOME>/sysman/prov/agentpush/2010-01-07_15-57-12-PM/logs/139.185.35.109

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The final screen shows you the results of the agent installation, including session information, the user inputs you provided, details of prerequisite checks, and installation details. Log locations are provided where relevant. Click **Done** when you finish reviewing the results.

Default Ports Used for Grid Control Installation



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

At installation time, the various ports required by the Grid Control infrastructure are assigned. The list of ports assigned during the installation process can be found in `<OMS_HOME>/cfgtoollogs/osmca/osmca_<timestramp>.log`. The default ports for Grid Control are listed in the slide, along with the recommended port ranges.

Configuring Firewalls

Before configuring your firewalls, consider these points:

- This should be the last stage of the deployment.
- If firewalls already exist, open the default Grid Control ports until installation and configuration is complete.
- Before you secure the management agents, test whether data is being uploaded to the repository.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Firewalls protect a company's IT infrastructure by providing the ability to restrict network traffic by examining each network packet and determining the appropriate course of action.

Firewall configuration typically involves restricting the ports that are available to one side of the firewall, for example, the internet. It can also be set up to restrict the type of traffic that can pass through a particular port such as HTTP. If a client attempts to connect to a restricted port (a port not covered by a security "rule") or uses a protocol that is incorrect, the client will be disconnected immediately by the firewall. Firewalls can also be used within a company intranet to restrict user access to specific servers.

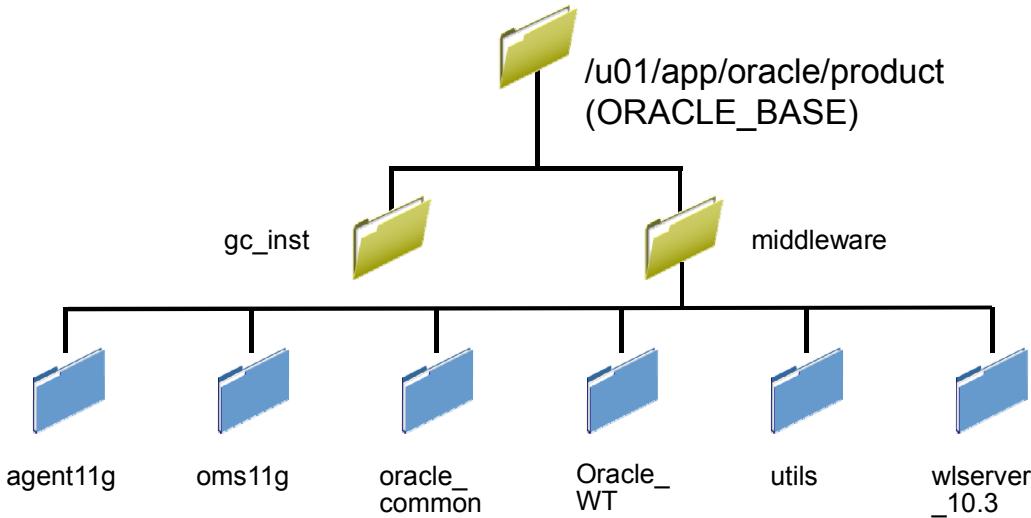
You can deploy the components of Grid Control on different hosts throughout your enterprise. These hosts can be separated by firewalls. Before configuring your firewall, consider the following:

- It should be the last phase of the Grid Control deployment. Make sure that all communication works up to this point before configuring firewalls into the architecture.
- If firewalls already exist in your environment, ensure that you open the default Grid Control communication ports until the installation and configuration processes are complete.

- When installing Grid Control, part of the configuration process is to restrict uploads from the management agents to only secure channels. Before performing this step, configure your firewall to allow HTTP and HTTPS traffic between the Oracle Management Agent and the Management Repository. Then, test by logging in to Grid Control and making sure that data is being uploaded to the Management Repository.

When real-time monitoring is initiated between Grid Control and a database target, a communication channel is opened directly between the OMS and the database target to allow Oracle Net Services traffic flow between them, bringing real-time metric information back. To make sure that this works without issue, ensure that the firewalls in the deployment environment allow communication between the OMS and database targets on the appropriate ports.

Grid Control Installation Directories



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

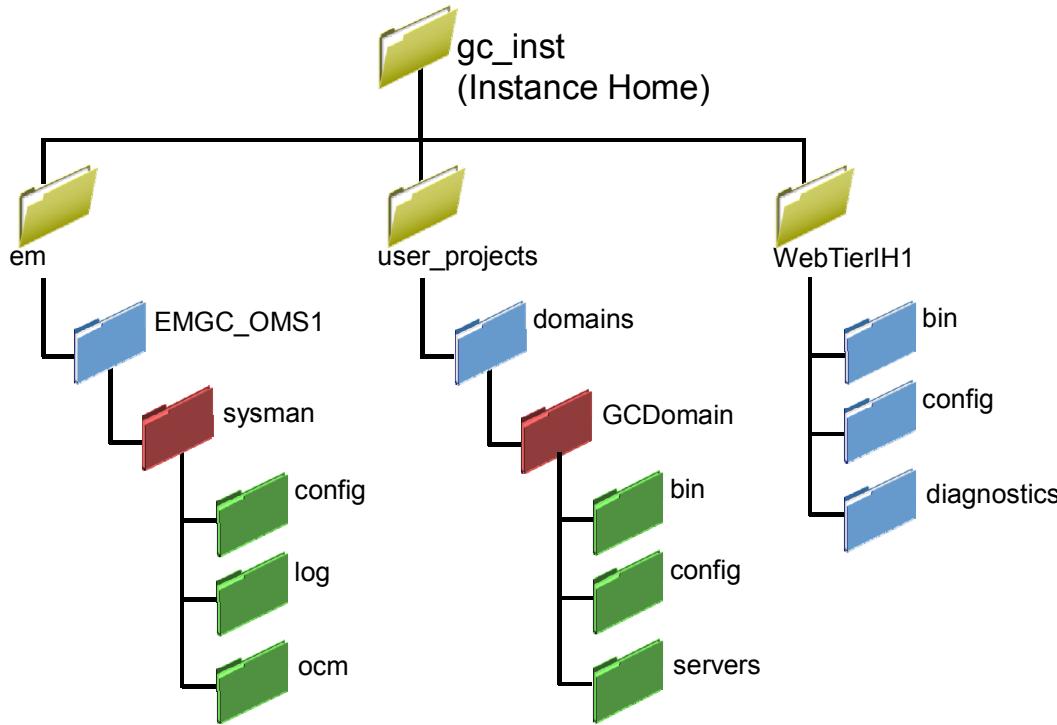
Before you perform maintenance and advanced configuration tasks, you must be familiar with the directories and files that are copied to disk when you install Grid Control. Understanding where specific files are located can help you if you need to troubleshoot installation or configuration problems.

When you install Grid Control, you specify a top-level directory for the software to be installed under, known as the Oracle base directory (for example, /u01/app/oracle/product in the slide). During the installation of the Oracle Management Service, a number of Oracle home directories are created:

- gc_inst : Run-time information for the Grid Control domain
- middleware/agent11g: ORACLE_HOME for the agent
- middleware/oms11g: ORACLE_HOME for the OMS
- middleware/oracle_common: Common user interface layer for applications
- middleware/Oracle_WT: Middleware WebTier home directory
- middleware/utils: Utilities for patching, and so on
- middleware/wlserver_10.3: The WebLogic Server home directory

Contents of these directories are covered in more detail in the next few slides.

Grid Control Installation Directories: Instance Home



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

During the Grid Control installation, an instance home is created (`gc_inst` in the slide) to separate the instance-specific configuration and log files from the read-only software. The instance home holds one of the most important files for the OMS, `emgc.properties`. This file contains configuration information such as port numbers, installation directories, and connection descriptions.

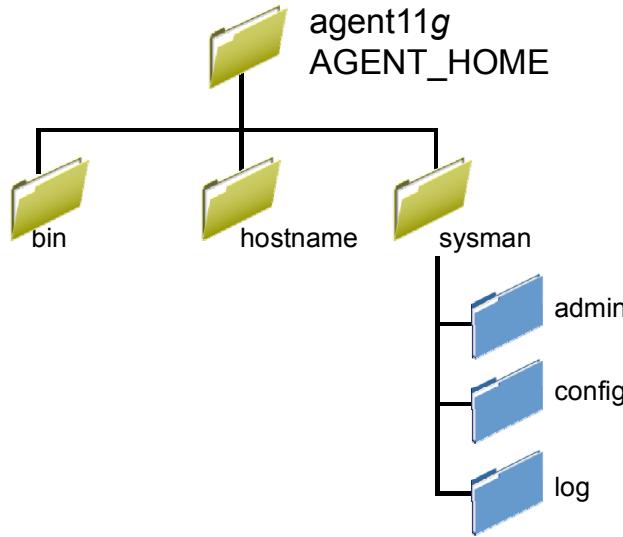
Some of the important directories for the instance home are:

- **Instance Home/em:** This directory stores the deployed Grid Control application files, including:
 - **EMGC_OMS1/sysman/config:** The `config` directory contains configuration files for online help, certificates used for OMS/agent communication, and placeholder properties files pointing users to `emctl` commands. **Note:** None of these files are end-user modifiable.
 - **EMGC_OMS1/sysman/log:** The `log` directory contains log and trace files for the OMS, including `emoms.log` and `emoms.trc`.
 - **EMGC_OMS1/sysman/ocm:** The `ocm` directory contains properties files used for passing OMS configuration properties to the OCM. **Note:** The files in this directory are not end-user modifiable.

- **Instance Home/user_projects:** This is a WLS artifact, which contains the WLS-specific configuration files under the domains directory for the Grid Control domain created when Grid Control was installed (GCDomain), including:
 - **domains/GCDomain/bin:** The bin directory contains files used to start the Grid Control domain.
 - **domains/GCDomain/config:** The config directory contains configuration files for the Grid Control domain.
 - **domains/GCDomain/servers:** The servers directory contains the managed server logs under servers/EMGC_OMS1/logs.
- **Instance Home/WebTier1H1:** This directory stores Apache-specific files, including:
 - **bin:** The bin directory contains the opmnctl utility, used to stop, start, and get process status for the process monitor and notification server.
 - **config:** The config directory contains Apache configuration files.
 - **diagnostics:** The diagnostics directory contains Apache diagnostic and log files

Note: Instance_home/user_projects/applications is still created but not listed in the slide because it is no longer used. It will be removed in a later release.

Grid Control Installation Directories: Agent



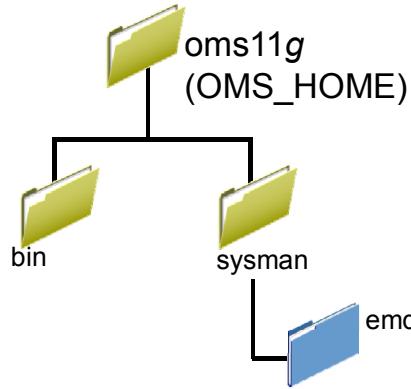
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The management agent is installed automatically when you install the Grid Control console. This local instance of the management agent gathers management information about the targets on the management service host. You can then manage those targets, such as the host itself, from the Grid Control console. The management agent home directory (AGENT_HOME in the slide) contains all the files required to configure and run the Oracle Management Agent on this host. The default AGENT_HOME location is at the same level as the middleware home (use the `emctl getemhome` command to check which home you are currently pointing to). Some of the important directories for the agent are:

- **AGENT_HOME/bin:** The `bin` directory contains many commands that control the Management Agent for this host, including the `emctl` command.
- **AGENT_HOME/hostname:** For Real Application Clusters, this directory contains all configuration, log files, and system files.
- **AGENT_HOME/sysman/admin:** This directory contains the files used by the management agent to define target types (such as databases, hosts, and so on), to run configuration scripts and other administrative tasks.
- **AGENT_HOME/sysman/config:** This directory contains the configuration files for the Oracle Management Agent. For example, this is where Grid Control stores the `emd.properties` file. The `emd.properties` file defines settings such as the Oracle Management Service upload URL for this particular agent.
- **AGENT_HOME/sysman/log:** This directory contains the log and trace files for the agent.

Grid Control Installation Directories: OMS



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

During the Grid Control installation, the static information for the OMS components is installed within the Oracle Management Service home directory (OMS_HOME in the slide).

Note: Remember this is in addition to the run-time information stored under the Instance Home.

Some of the important directories for the OMS_HOME are:

- **OMS_HOME/bin:** The `bin` directory in the management service home contains commands used to control the components of the Grid Control installation.
- **OMS_HOME/sysman:** The `sysman` directory in the management service home contains the system management files associated with this Grid Control installation.
- **OMS_HOME/sysman/emd:** The `emd` directory contains information about targets discovered on hosts.

Quiz

Identify the file that provides a list of the ports used in an interactive Grid Control installation.

- a. <OMS_HOME>/cfgtoollogs/omsca/omsca_<timesta
mp>.log
- b. <OMS_HOME>/install/portlist.ini
- c. <OMS_HOME>/install/staticports.ini



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned how to:

- Install Grid Control
- Describe the installation process for the various components that make up Grid Control
- Discuss the ports used for Grid Control installation
- Explain the directory structure of Grid Control



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Setting Up Enterprise Manager Grid Control

9

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Configure Enterprise Manager Grid Control to set up additional administrators
- Identify the types of privileges used in Grid Control
- Use roles to assign privileges to groups of administrators
- Set up preferred credentials for simplifying access to managed targets



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Grid Control Administrators

Administrators are users defined in Grid Control to perform management tasks. Grid Control enables you to create two types of accounts:

- The super administrator account
- The administrator account



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

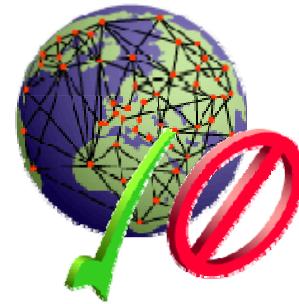
Administrators are users defined in the Management Repository of Grid Control that enable you to perform administrative tasks. You can set the roles and privileges of the administrators. Grid Control enables you to create two types of accounts:

- **The super administrator account:** Grid Control is installed with a default super administrator account named sysman (**Note:** sys and system are also defined as super administrator accounts). This super administrator account cannot be deleted or renamed, because it is also the repository owner. The super administrator can manage all other administrator accounts and set up all administrator credentials. The super administrator can:
 - Create Grid Control privileges and roles
 - Perform the initial setup of Grid Control (for example, defining email configurations and defining notifications methods)
 - Add targets to Grid Control
 - Perform any action on any target in the system
- **The administrator account:** Administrator accounts provide users permission to perform administrative tasks and access administrative information. You can set up each administrator account with:
 - Password configuration
 - Email address
 - System and target privileges
 - Role, or set of privileges

Privileges

There are three categories of privileges in Grid Control:

- System privileges
- Target privileges
- Object privileges



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Privileges are the right to perform certain management actions within Grid Control. These privileges are divided into three categories:

- **System privileges:** Rights to perform systemwide operations within Grid Control (for example, view any target in the system)
- **Target privileges:** Rights to perform operations on a target (for example, view properties and monitor information about a specific target)
- **Object privileges:** Rights to perform operations on objects such as jobs and reports

Note: Object privileges are granted via the Access link on the object home page. Because they are not part of user or role creation, they will not be discussed in this lesson. See the lesson titled “Using the Job System” for more details.

System Privileges

System privileges allow a user to perform systemwide operations. They are:

- JVM DIAGNOSTICS ADMINISTRATOR
- JVM DIAGNOSTICS USER
- REQUEST MONITORING ADMINISTRATOR
- REQUEST MONITORING USER
- PUBLISH REPORT
- USE ANY BEACON
- ADD ANY TARGET
- VIEW ANY TARGET
- CREATE PRIVILEGE PROPAGATING GROUP
- MONITOR ENTERPRISE MANAGER



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

System privileges provide the ability to perform systemwide operations. These system privileges are:

- **JVM DIAGNOSTICS ADMINISTRATOR:** Allows the administrator to manage Java Diagnostics administrative operations
- **JVM DIAGNOSTICS USER:** Allows the administrator to view Java Diagnostics data
- **REQUEST MONITORING ADMINISTRATOR:** Allows the administrator to manage request data operations in the new Request Monitor console
- **REQUEST MONITORING USER:** Allows the administrator to view request data in the new Request Monitor console
- **PUBLISH REPORT:** Allows the administrator to publish reports for public viewing
- **USE ANY BEACON:** Allows the administrator to use any beacon (an application to monitor transactions from different user communities or geographical regions) on any monitored host to monitor transactions, URLs, and network components
- **ADD ANY TARGET:** Allows the administrator to add any target to Grid Control for monitoring, administration, and management
- **VIEW ANY TARGET:** Allows the administrator to view any target on the system, including Oracle Management Agents and Management Services. Whenever the **VIEW ANY TARGET** privilege is granted, the **MONITOR ENTERPRISE MANAGER** privilege is also granted by default.

- **CREATE PRIVILEGE PROPAGATING GROUP:** Allows the administrator to create privilege propagating groups. Privileges granted on a privilege propagating group are automatically granted on the members of the group.
- **MONITOR ENTERPRISE MANAGER:** Allows the administrator to monitor the availability and performance of Grid Control itself, and grants the administrator access to the following targets: the management repository, the management service, and all Oracle Management Agents in the global enterprise

Target Privileges

Target privileges allow a user to perform operations on a target.
They are:

- VIEW
- OPERATOR
 - BLACKOUT TARGET
 - MANAGE TARGET METRICS
 - CONFIGURE TARGET
 - MANAGE TARGET ALERTS
- FULL



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Target privileges allow a user to perform operations on a target within Grid Control. The target privileges available are:

- **VIEW:** Allows the administrator to see the target's properties, inventory, and monitor information about a target
- **OPERATOR:** Allows the administrator to perform common administrative functions or tasks on the targets within Grid Control. It has the following subprivileges:
 - **BLACKOUT TARGET** allows the administrator to create, edit, schedule, and stop blackouts (a tool for suspending any data collection activity) on a target.
 - **MANAGE TARGET METRICS** allows the administrator to edit thresholds for metric and policy settings, apply monitoring templates, and manage user-defined metrics.
 - **CONFIGURE TARGET** allows the administrator to edit target properties and modify monitoring configurations.
 - **MANAGE TARGET ALERTS** allows the administrator to clear stateless alerts, manually reevaluate alerts and acknowledge alerts for the target.
- **FULL:** Gives the administrator all the above target privileges including the ability to delete a target and configure credentials for the maintenance operations of a target. Granting database full privileges to the user automatically grants host view privileges as well.

Granting any privileges on a target automatically grants the `VIEW` target privilege on that target's host. System privileges will override the target privileges for a target.

Note: Target privileges provide rights to manage that target within Grid Control; they do not provide any additional privileges within the target itself.

Roles

Roles enable you to grant a set of predefined privileges to a group of administrators. Roles can be based on:

- Geographical locations
- Line of business
- Any other IT model



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can create roles in Grid Control to group system and target privileges and grant these to other administrators or to other roles. Using Roles and privileges, you can manage security across all functional areas of Grid Control. Roles can be based on:

- **Geographic location:** For example, you can define a role for UK administrators to manage UK systems.
- **Line of business:** For example, you can define a role for administrators of the human resource systems or payroll systems.
- **Any other IT model**

Using roles you can, if required, update the privileges across all administrators. An easy way to do this is through the use of the PUBLIC role, an out-of-the-box role that has no privileges. You can use the PUBLIC role to assign a privilege to all administrators in Grid Control because by default, all Grid Control administrators have the PUBLIC role.

Note: Roles defined in Grid Control are not the same as database roles.

Creating a Super Administrator Account

Enterprise Manager Configuration | Management Services and Repository | Agents

Overview of Setup

Setup allows you to access general Enterprise Manager configuration and system monitoring functions. Depending on the system and target privileges that have been granted, you can access setup functions for the following administrative area(s):

Enterprise Manager Configuration: lets you perform administrative operations such as adding new Administrators, managing Monitoring Templates, and establishing Blackouts. Your administrator privileges determine which configuration operations are displayed. See [Introduction to Setting Up Enterprise Manager](#) for more information.

Management Services and Repository: lets you monitor system performance and access diagnostic information for the Oracle Management Services and Management Repository. You can view:

- The overall health of Enterprise Manager.
- The status and performance of the Repository DBMS Jobs that handle Enterprise Manager's maintenance and monitoring functionality.
- The health and configuration of all Management Services.
- Performance errors for the DBMS jobs and Management Service components (Repository Metrics).

See [Monitoring The Management System](#) for more information.

Agents: lets you view general configuration, status, and performance information of the Oracle Management Agents that have been installed and configured for managed hosts. See [About Oracle Management Agents](#) for more information.

Only Super Administrators can access Setup functions for all administrative areas.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To create a new Super Administrator, perform the following steps:

1. On the Grid Control Home page, click **Setup** (at the upper-right corner of the page).
2. Click **Administrators** (in the left Navigator).

Creating a Super Administrator Account

The screenshot shows two overlapping windows. The top window is titled 'Administrators' and contains a list of existing administrators: SYS, SYSMAN, and SYSTEM. A red circle labeled '3' highlights the 'Create' button in the top right corner of this window. The bottom window is titled 'Create Administrator: Properties' and is used to create a new administrator named 'Super_Admin'. It includes fields for Name, Password, Confirm Password, Password Profile (set to 'DEFAULT'), and E-mail Address. A checkbox labeled 'Super Administrator' is checked and highlighted with a red box. A red circle labeled '4' highlights this checkbox. The Oracle logo is visible at the bottom right of the interface.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

3. Click **Create**.
4. Enter a name and password. Select Super Administrator, and enter other details as appropriate. Click **Next**.
5. Click **Finish**.

Note: Only a Super Administrator can create a Super Administrator account in Grid Control.

Creating an Administrator Account

The screenshot shows two sequential screens for creating an administrator account:

Step 1: Create Administrator: Properties

- Name: DB_Administrator
- Password: (redacted)
- Confirm Password: (redacted)
- Password Profile: DEFAULT
- E-mail Address: (redacted)
- Description: (redacted)
- Checkboxes: Prevent password change, Expire password now
- Buttons: View, Manage Profiles, Super Administrator

Step 2: Create Administrator DB_Administrator: Roles

Available Roles: PUBLIC

Selected Roles: DB_ROLE

Navigation: Properties, Roles, System Privileges

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To create an administrator, click **Setup** from the Grid Control Home page as before, then click **Administrators** in the left navigator, and click **Create**. Then perform the following steps:

1. Enter a name and password, and then click **Next**.
2. If desired, select a Role and click **Move**. Then click **Next**.

Creating an Administrator Account

3

Create Administrator DB_Administrator: System Privileges

Select the System Privileges that you want to grant to this Enterprise Manager Administrator. System Privileges give the administrator the right to perform particular management actions.

Select	Name	Description
<input type="checkbox"/>	JVM Diagnostics Administrator	Able to manage JVM Diagnostics admin operations
<input type="checkbox"/>	JVM Diagnostics User	View JVM Diagnostics Data
<input type="checkbox"/>	Request Monitoring Administrator	Able to manage Request Monitoring admin operations
<input type="checkbox"/>	Request Monitoring User	View Request Monitoring data
<input type="checkbox"/>	Publish Report	Ability to publish reports for public viewing
<input type="checkbox"/>	Use any Beacon	Use any Beacon on any monitored host to monitor transactions, URLs, and network components. Beacon is installed with the Oracle Agent.
<input type="checkbox"/>	Add any Target	Add any target in Enterprise Manager
<input type="checkbox"/>	View any Target	View all discovered targets in the environment (including Agents, and Management Service and Repository targets). This system privilege automatically includes the MONITOR ENTERPRISE MANAGER system privilege.
<input type="checkbox"/>	Create Privilege Propagating Group	Ability to create privilege propagating groups.Privileges granted on a privilege propagating group will be automatically granted on the members of the group
<input type="checkbox"/>	Monitor Enterprise Manager	Monitor Enterprise Manager performance. Performance is determined by how efficiently the Management Services and their components are running and processing, and how well the DBMS jobs handling the maintenance and monitoring of Enterprise Manager are running.

[Cancel](#) [Back](#) [Step 3 of 5](#) [Next](#) [Review](#)

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

3. Select the system privileges that you want to include for this administrator, and then click **Next**.

Creating an Administrator Account

Create Administrator DB_Administrator: Target Privileges 4

Target Privileges give the Administrator the right to perform particular actions on targets. Table below shows privileges on the targets which would be given to the Administrator.

Use "Batch Assign" button to assign privileges to multiple targets. Privileges for the selected targets will be replaced by the batch settings. To edit individual target, click "Edit" icon.

Name	Type	Privilege
db56.us.oracle.com		

Select All | Select None

Select Name

Create Administrator DB_Administrator: Target Privileges 5

Select the Target Privileges that you want to grant to this Enterprise Manager administrator. Target Privileges give the administrator the right to perform particular actions on targets selected on the previous page.

- Full
Ability to do all operations on the target, including delete the target
- Operator
Ability to do normal administrative operations on the target, such as configure a blackout and edit the target properties
- Blackout Target
Ability to create, edit, schedule and stop a blackout on the target
- Manage Target Metrics
Ability to edit threshold for metric and policy setting, apply monitoring templates, and manage User Defined Metrics
- Configure Target
Ability to edit target properties and modify monitoring configuration

Cancel **Continue**

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

4. Select the targets that you want this administrator to have access to, and then click **Next** (if there are no targets available, click the **Add** button to select targets)
Note: Selecting a group automatically includes all targets in that group.
5. The default privilege listed is **VIEW**. Click the pencil icon next to **VIEW** to select other privileges. Select any target privileges that you want the administrator to have from the previous page, and then click **Next**.
6. Click **Finish** to create the administrator account.

Maintaining Administrators

It is possible to maintain the properties of administrators, such as:

- Changing the password or email address
- Assigning or changing:
 - Roles
 - System privileges
 - Targets
 - Target privileges

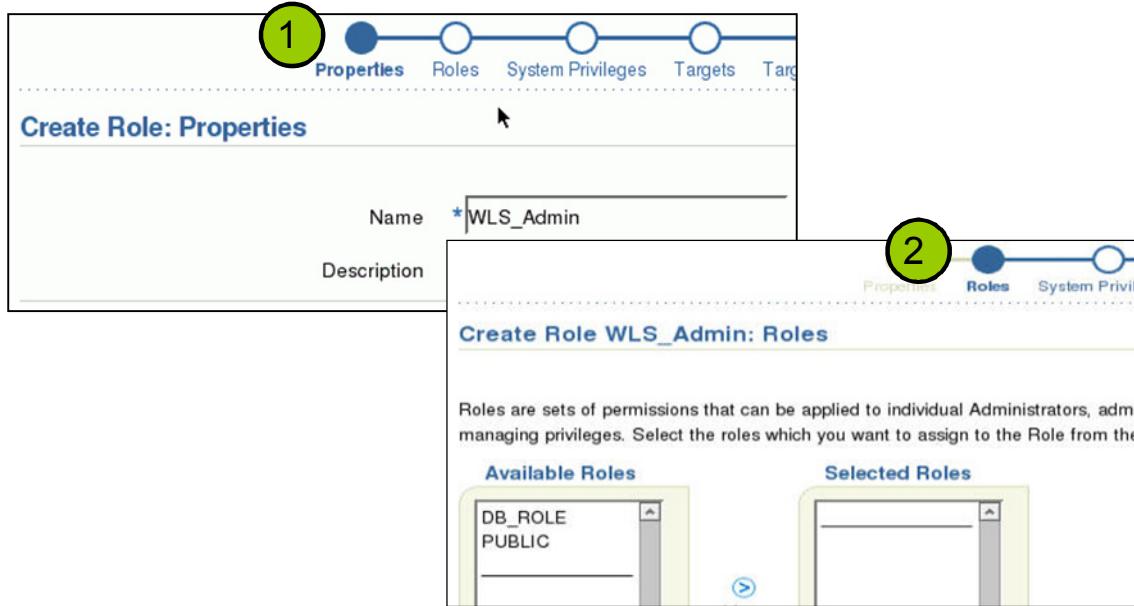


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

With the same setup screen that you use to create administrators, you can also maintain existing administrators. Select the administrator that you want to edit (use the search functionality if necessary), and click **Edit** or just click the name of the administrator. This presents a step-by-step interface similar to what you see when creating administrators. Make any changes you need, and when you reach the Review screen, click **Finish** to save your changes.

Creating Roles

On the Grid Control Home page, click Setup, and then click Roles.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Roles are created from the Setup page of the Grid Control console. When you click Roles on the Setup page, the roles that are currently defined are displayed. You administer the roles in your Grid Control environment from here. There is a search mechanism to find a specific role. For the creation of new roles, there is a Create button and a Create Like button. If you want to create a role that has the same makeup as an existing role, click Create Like. If you want to create a role from the beginning, click Create. This displays the first step in creating a role, where you specify the name for the role and a description. When that is complete, click Next to take you through the remaining setup screens. The process has seven steps:

1. Enter the name and description of the role.
2. Include any existing roles (new role inherits the features of chosen roles).

Creating Roles

3

Select	Name	Description
<input type="checkbox"/>	JVM Diagnostics Administrator	Able to manage JVM Diagnostics admin operations
<input type="checkbox"/>	JVM Diagnostics User	View JVM Diagnostics Data
<input type="checkbox"/>	Request Monitoring Administrator	Able to manage Request Monitoring admin operations
<input type="checkbox"/>	Request Monitoring User	View Request Monitoring data
<input type="checkbox"/>	Publish Report	
<input type="checkbox"/>	Use any Beacon	
<input type="checkbox"/>	Add any Target	
<input type="checkbox"/>	View any Target	
<input type="checkbox"/>	Create Privilege Propagating Group	
<input type="checkbox"/>	Monitor Enterprise Manager	

4

Create Role WLS_Admin: Target Privileges			
Target Privileges give the Administrator the right to perform particular actions on targets. Table below shows privileges on the targets which would be granted to the Role			
Use "Batch Assign" button to assign privileges to multiple targets. Privileges for the selected targets will be replaced by the batch settings. To edit individual privileges use the "Edit" icon.			
Name <input type="text"/> Type <input type="button" value="Go"/> <input type="button" value="Clear"/> <input type="button" value="Remove"/> <input type="button" value="Batch Assign"/> <input type="button" value="Add"/> <input type="button" value="Bulk Assign"/>			
Select All Select None			
Select	Name	Type	privilege
<input checked="" type="checkbox"/>	/secFarm_GCDomain/GCDomain	Oracle WebLogic Domain	<input type="button" value="Edit"/>
<input checked="" type="checkbox"/>	/secFarm_GCDomain/GCDomain/EMGC_ADMINSERVER	Oracle WebLogic Server	<input type="button" value="Edit"/>
<input type="checkbox"/>	/secFarm_GCDomain/GCDomain/EMGC_ADMINSERVER/FMW_Welcome Page Application(11.1.0.0.0)	Application Deployment	<input type="button" value="Edit"/>
<input type="checkbox"/>	/secFarm_GCDomain/GCDomain/EMGC_ADMINSERVER/mds-sysman_nds	Metadata Repository	<input type="button" value="Edit"/>
<input checked="" type="checkbox"/>	/secFarm_GCDomain/GCDomain/EMGC_OMS1	Oracle WebLogic Server	<input type="button" value="Edit"/>



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

3. Choose the desired system privileges.
4. Include any targets that the administrators who have this role should have access to.

Creating Roles

5

Select the Target Privileges that you want to grant to this Enterprise Manager administrator. Target Privileges give the administrator the right to perform particular actions on targets selected on the previous page.

Full Ability to do all operations on the target, including delete the target
 Operator Ability to do normal administrative operations on the target, such as configure a blackout and edit the target properties
 Blackout Target Ability to create, edit, schedule and stop a blackout on the target
 Manage Target Metrics Ability to edit threshold for metric and policy setting, apply monitoring templates, and manage User Defined Metrics
 Configure Target Ability to edit target properties and modify monitoring configuration
 Manage Target Alerts Ability to clear stateless alerts, manually re-evaluate alerts and acknowledge alerts
 View Ability to view properties, inventory and monitor information about a target

6

Select the administrators that you want to grant this Enterprise Manager role to.

Available Administrators	Selected Administrators
DB_ADMINISTRATOR	

Available Administrators: DB_ADMINISTRATOR
Selected Administrators: None

Move Move All

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

5. For each of these targets, decide the target-level privileges this role should have.
6. Select the administrators you want to grant this role to. (This can also be performed later.)
7. Review and click **Finish** when complete.

Preferences

Use the Preferences screen to administer the preferences for the current user:

- General
- Preferred Credentials
- Notification:
 - My Rules
 - Public Rules
 - Schedule
- Target Subtabs
- Accessibility



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

From any page in Grid Control, you can access the Preferences screen for the current user by clicking the Preferences link at the upper-right corner of the page. On the Preferences screen, you can define and maintain the preferences for the current user. Preferences are divided into five categories:

- **General:** Change the password and maintain email information.
- **Preferred Credentials:** Set login credentials for the targets you manage.
- **Notification:**
 - **My Rules:** Choose targets and conditions for which you want to receive notifications.
 - **Public Rules:** View and subscribe to rules that other users have created.
 - **Schedule:** Define a schedule to control when you receive notifications based on your rules.
- **Target Subtabs:** Customize the target subtabs you see on the main Targets page.
- **Accessibility:** Tell Grid Control you use a screen reader so accessibility-specific constructs are added.

Notification information is covered in the lesson titled “Managing Systems and Services.”

Note: If you are logged in as a Super Administrator, you see the Rules and Schedule subcategories under Notification. Only regular administrators see My Rules and Public Rules.

Preferred Credentials

Preferred credentials simplify access to targets that you manage. For example, you can set preferred credentials for target types such as:

- Databases
- Host
- Listener



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Preferred credentials simplify access to managed targets by storing target login credentials in the management repository. With preferred credentials set, users can access a Grid Control target that recognizes those credentials without being prompted to log in to the target. Preferred credentials are set on a per-user basis, thus ensuring the security of the managed enterprise environment. There are no preferred credentials set by default.

Preferred credentials in Grid Control have the following characteristics:

- Normal database credentials do not require SYSDBA privileges. These credentials are the default login credentials you use when accessing a Grid Control function that requires a database connection.
- You use SYSDBA database credentials for privileged functions that access non-open databases or perform database startup and shutdown operations.
- You use host credentials to execute remote operations and jobs that run applications, which access a database (for example, SQL*Plus).

Setting Preferred Credentials

Preferred credentials can be set, modified, or deleted from the Preferred Credentials page.

Target Type	Total Targets	Targets with Credentials Set	Default Credentials Set	Set Credentials
Listener	1	0	No	
CSA Collector	1	0	No	
Host	1	0	No	
Database Instance	1	0	No	
Agent	1	0	No	

TIP You can set default credentials for each target type. Default credentials are used for any targets that do not have preferred credentials explicitly set.

My Oracle Support Preferred Credentials

This is required to connect to My Oracle Support to search knowledge, file service requests and work with patches and updates.

[Set Credentials](#)

[Home](#) | [Targets](#) | [Deployments](#) | [Alerts](#) | [Compliance](#) | [Jobs](#) | [Reports](#) | [My Oracle Support](#) | [Setup](#) | [Preferences](#) | [Help](#) | [Logout](#)

Copyright © 1996, 2010, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its affiliates.
Other names may be trademarks of their respective owners.
[About Oracle Enterprise Manager](#)



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To set preferred credentials, perform the following steps:

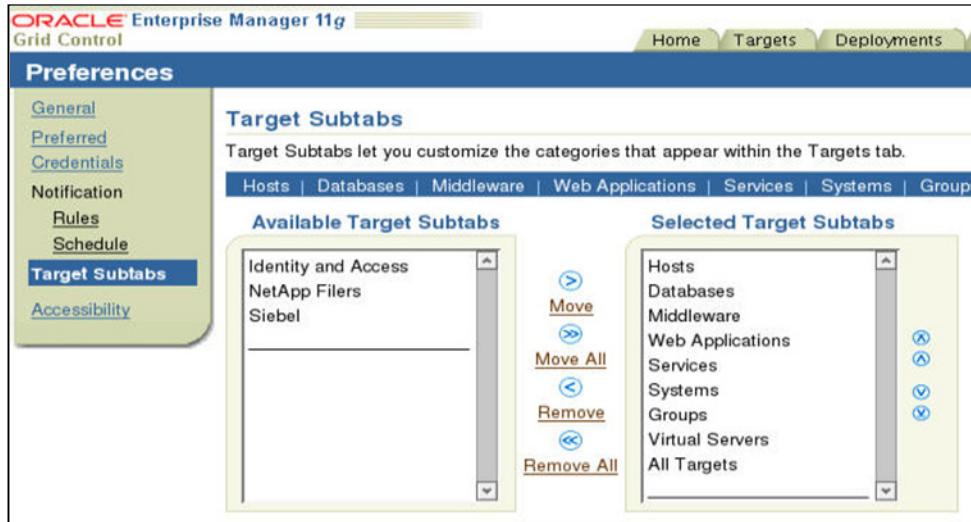
1. On the Preferences page, select Preferred Credentials in the navigation pane.
2. Click the **Set Credentials** icon next to each target type.

To set the preferred credentials for a database target, you enter Normal, SYSDBA, or Host login credentials (usernames and passwords).

If your organization uses standardized usernames and passwords, you can enter default credentials that will be used for all targets. Alternatively, you can enter usernames and passwords for each individual target. If you enter both default and individual target credentials, the individual target credentials override the default credentials. After you enter the preferred credentials values, click **Apply**. You can also click **Test** to make sure that the credentials are set appropriately.

Managing Target Subtabs

You can customize Grid Control Target Subtabs to match your division of system responsibility.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Target Subtabs enable you to customize the categories that appear within the Targets tab. By default, Hosts appears first. For example, if you are not interested in middleware, you can remove the Middleware subtab. As with preferred credentials, the modifications made to the target subtabs are set on a per-user basis.

To modify target subtabs, perform the following steps:

1. On the Preferences page, click **Target Subtabs** from the navigation pane.
2. Add or remove the targets as desired.
3. Reorder the targets in the Selected Target Subtabs list by selecting the target and clicking the up or down arrows.
4. Click **Apply**. The next time you click the Targets tab, the subtabs shown will reflect the changes you just made.

Note: Groups are covered in the lesson titled “Managing Groups.”

Enterprise Manager Command Line Interface

The Enterprise Manager Command Line Interface (EM CLI) provides text-based access to the Grid Control functionality. It provides:

- Integration with third-party or custom software via scripting
- Management Repository data extraction for manipulation and reporting purposes



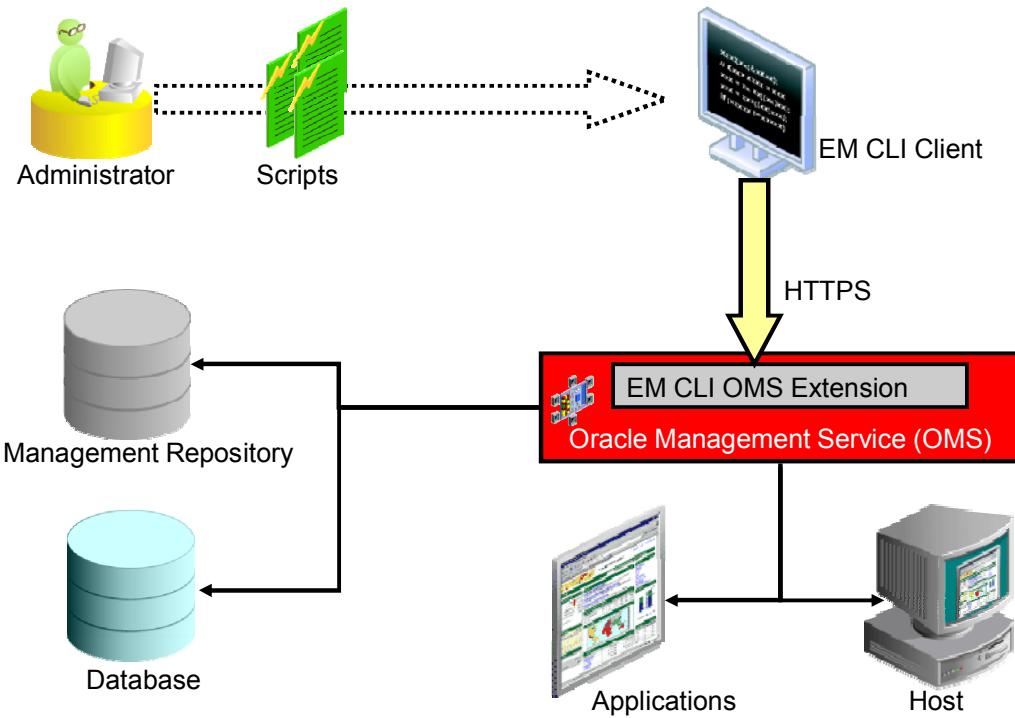
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The EM CLI enables you to access the Grid Control functionality from text-based consoles, such as command windows and shells. This provides the ability to use custom scripts (such as SQL*Plus, PERL, Tcl, and OS Shell) making integration with third-party or custom software possible. It also enables you to extract data from the management repository for manipulation or reporting purposes.

You can access the EM CLI kit from the following location:

`OMS_host.domain:<port>/em/console/emcli/download`

Setting Up EM CLI



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

EM CLI consists of two components:

- **EM CLI Client:** The EM CLI Client is a Java application that accepts a command as input. The EM CLI Client then uses the input command to identify a verb to execute the command. A verb is a Java plug-in extension to the EM CLI Client. A verb services the command with its specific options and posts the results to the standard output stream. Any errors are posted to the error output stream. The verb also returns an integer exit value that the EM CLI Client sets as the exit value of the command in the Client's calling environment (the operating system console).
- **EM CLI Oracle Management Service Extension (EM CLI OMS Extension):** The EM CLI OMS Extension is a standard Grid Control console page installed in the OMS. The EM CLI OMS Extension also uses the input command to identify a verb to execute a command.

A verb can be executed locally or remotely. A remote verb connects to EM CLI OMS Extension in the Grid Control OMS Console through HTTP/HTTPS and sends the command line through HTTP to the OMS for processing. The remote verb can access the management repository or the management agent through OMS services. The remote verb uses the Grid Control user credentials to log in to access OMS.

The EM CLI OMS Extension is automatically installed when you install the OMS. However, you need to install and set up the EM CLI client. The EM CLI Client kit (`emclikit.jar`) can be accessed through `HTTPS: OMS_host.domain:<port>/em/console/emcli/download`. The `emclikit.jar` file is physically located in the `$ORACLE_HOME/sysman/jlib` directory of the Grid Control OMS home.

To set up the EM CLI Client, perform the following steps:

1. Set the `JAVA_HOME` environment variable and add the following to the PATH:
 - `export JAVA_HOME =<full_path_to_your_Java_install>`
 - `export PATH=$JAVA_HOME/bin:$PATH`
2. Install the EM CLI Client. Run the following command:
 - `java -jar emclikit.jar client -install_dir=<emcli client dir>`
3. From the directory where you installed EM CLI, run:
 - `emcli setup -url=http://OMS_host.domain:<port>/em - username=em_user`
 - You are prompted at the command line to enter the password when connecting to the EM CLI Management Services. Running the `setup` verb installs all available verbs-associated command-line help from the EM CLI Management Service. You must run `setup` each time you want to connect to a different OMS.

Note: For more information about setting up EMCLI and using verbs, refer to the Grid Control documentation set.

Quiz

Who can create a Super Administrator account in Grid Control?

- a. Administrator
- b. Super Administrator
- c. sysman



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Answer: b, c

Quiz

Which other privilege is granted when you grant the VIEW ANY TARGET privilege to an Administrator?

- a. ADD ANY TARGET
- b. USE ANY BEACON
- c. PUBLISH REPORT
- d. MONITOR ENTERPRISE MANAGER



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Answer: d

Quiz

Which target privilege enables a user to start a blackout on a target, but not delete a target?

- a. VIEW
- b. OPERATOR
- c. FULL



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Answer: b

Summary

In this lesson, you should have learned how to:

- Configure Enterprise Manager Grid Control to set up additional administrators
- Identify the types of privileges used in Grid Control
- Use roles to assign privileges to groups of administrators
- Set up preferred credentials for simplifying access to managed targets



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

10

Introduction to Oracle Data Guard

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

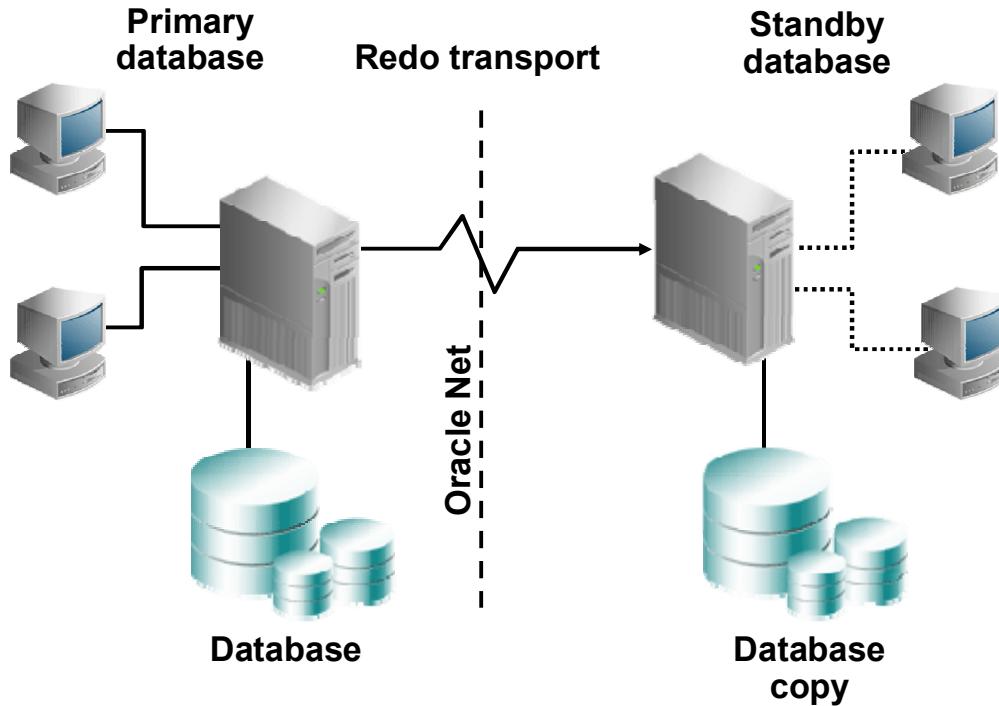
After completing this lesson, you should be able to do the following:

- Describe the basic components of Oracle Data Guard
- Explain the differences between physical and logical standby databases
- Explain the benefits of implementing Oracle Data Guard



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

What Is Oracle Data Guard?



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Data Guard is a central component of an integrated Oracle Database High Availability (HA) solution set that helps organizations ensure business continuity by minimizing the various kinds of planned and unplanned down time that can affect their businesses.

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.

All Data Guard standby databases can enable up-to-date read access to the standby database while redo being received from the primary database is applied. This makes all standby databases excellent candidates for relieving the primary database of the overhead of supporting read-only queries and reporting.

Types of Standby Databases

- Physical standby database
 - Identical to the primary database on a block-for-block basis
 - Synchronized with the primary database through application of redo data received from the primary database
 - Can be used concurrently for data protection and reporting
- Logical standby database
 - Shares the same schema definition
 - Synchronized with the primary database by transforming the data in the redo received from the primary database into SQL statements and then executing the SQL statements
 - Can be used concurrently for data protection, reporting, and database upgrades



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Physical Standby Database

A physical standby database is physically identical to the primary database, with on-disk database structures that are identical to the primary database on a block-for-block basis. The physical standby database is updated by performing recovery using redo data that is received from the primary database. Oracle Database 11g enables a physical standby database to receive and apply redo while it is open in read-only mode.

Logical Standby Database

A logical standby database contains the same logical information (unless configured to skip certain objects) as the production database, although the physical organization and structure of the data can be different. The logical standby database is kept synchronized with the primary database by transforming the data in the redo received from the primary database into SQL statements and then executing the SQL statements on the standby database. This is done with the use of LogMiner technology on the redo data received from the primary database. The tables in a logical standby database can be used simultaneously for recovery and for other tasks such as reporting, summations, and queries.

Note: For more information about LogMiner, see *Oracle Database Utilities* in the Oracle Database 11g documentation.

Types of Standby Databases

- Snapshot standby database
 - Fully updatable standby database
 - Created by converting a physical standby database
 - Local updates are discarded when a snapshot standby database is converted back into a physical standby database.
 - Can be used for testing



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Snapshot Standby Database

A snapshot standby database is a database that is created by converting a physical standby database into a snapshot standby database. The snapshot standby database receives redo from the primary database, but does not apply the redo data until it is converted back into a physical standby database. The snapshot standby database can be used for updates, but those updates are discarded before the snapshot standby database is converted back into a physical standby database. The snapshot standby database is appropriate when you require a temporary, updatable version of a physical standby database.

Types of Data Guard Services

Data Guard provides three types of services:

- Redo transport services
- Apply services
 - Redo Apply
 - SQL Apply
- Role management services



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following types of services are available with Data Guard:

- **Redo transport services:** Control the automated transmittal of redo information from the primary database to one or more standby databases or archival destinations.
- **Apply services:** Control when and how data is applied to the standby database.
 - **Redo Apply:** Technology used for physical standby databases. Redo data is applied on the standby database by using Oracle media recovery.
 - **SQL Apply:** Technology used for logical standby databases. The received redo data is first transformed into SQL statements, and then the generated SQL statements are executed on the logical standby database.
- **Role management services:** A database operates in one of two mutually exclusive roles: primary or standby. Role management services operate in conjunction with redo transport services and apply services to change these roles dynamically as a planned transition (called a *switchover operation*) or as a result of database failure due to a failover operation.

Role Transitions: Switchover and Failover

Oracle Data Guard supports two role-transition operations:

- **Switchover**
 - Planned role reversal
 - Used for OS or hardware maintenance
- **Failover**
 - Unplanned role reversal
 - Emergency use
 - Zero or minimal data loss (depending on choice of data protection mode)
 - Can be initiated automatically when fast-start failover is enabled



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

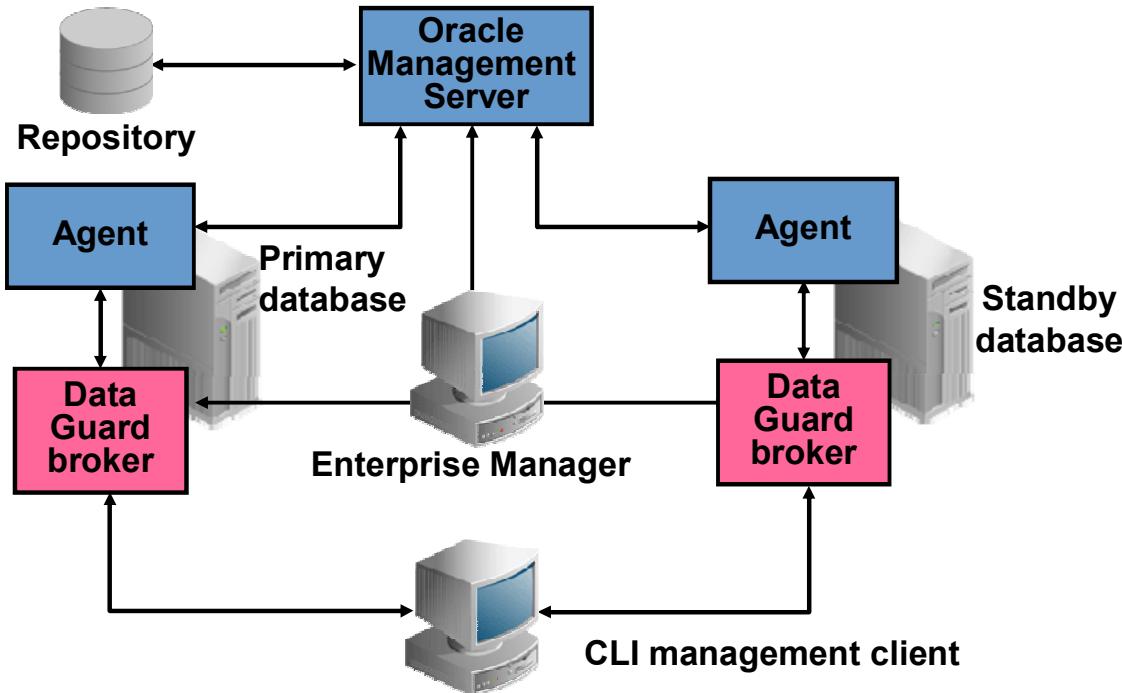
Data Guard enables you to change the role of a database dynamically by issuing SQL statements or by using either of the Data Guard broker's interfaces. Data Guard supports two role-transition operations:

- **Switchover:** The switchover feature enables you to switch the role of the primary database to one of the available standby databases. The chosen standby database becomes the primary database, and the original primary database then becomes a standby database.
- **Failover:** You invoke a failover operation when a catastrophic failure occurs on the primary database. During a failover operation, the failed primary database is removed from the Data Guard environment, and a standby database assumes the primary database role. You invoke the failover operation on the standby database that you want to fail over to the primary role. You can also enable fast-start failover, which allows Data Guard to automatically and quickly fail over to a previously chosen synchronized standby database.

Databases that are disabled after a role transition are not removed from the broker configuration, but they are disabled in the sense that the databases are no longer managed by the broker. To reenable broker management of these databases, you must reinstate or re-create the databases.

Note: See the lesson titled “Performing Role Transitions” for detailed information.

Oracle Data Guard Broker Framework



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Data Guard Broker

The Oracle Data Guard broker is a distributed management framework that automates and centralizes the creation, maintenance, and monitoring of Data Guard configurations. After creating the Data Guard configuration, the broker monitors the activity, health, and availability of all systems in the configuration.

Choosing an Interface for Administering a Data Guard Configuration

- Data Guard broker configuration:
 - DGMGRL command-line interface
 - Enterprise Manager Grid Control
 - SQL commands to query data dictionary views
- Non–Data Guard broker configuration:
 - SQL commands



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

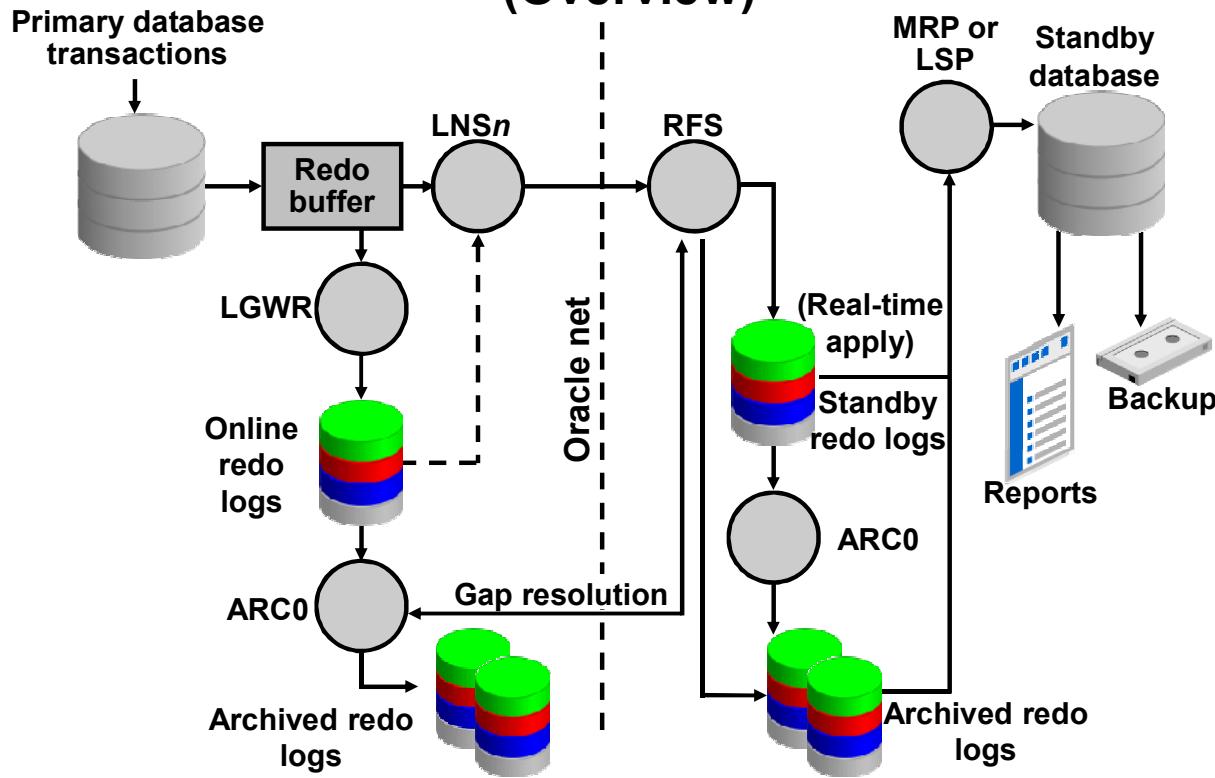
You can use Oracle Enterprise Manager Grid Control or the Data Guard broker's own specialized command-line interface (DGMGRL) to take advantage of the broker's management capabilities.

Enterprise Manager Grid Control provides a web-based interface that combines with the broker's centralized management and monitoring capabilities so that you can easily view, monitor, and administer primary and standby databases in a Data Guard configuration.

You can also use DGMGRL to control and monitor a Data Guard configuration. You can perform most of the activities that are required to manage and monitor the databases in the configuration from the DGMGRL prompt or in scripts.

If you do not create a Data Guard broker configuration, you can manage your standby databases by using SQL commands.

Oracle Data Guard: Architecture (Overview)



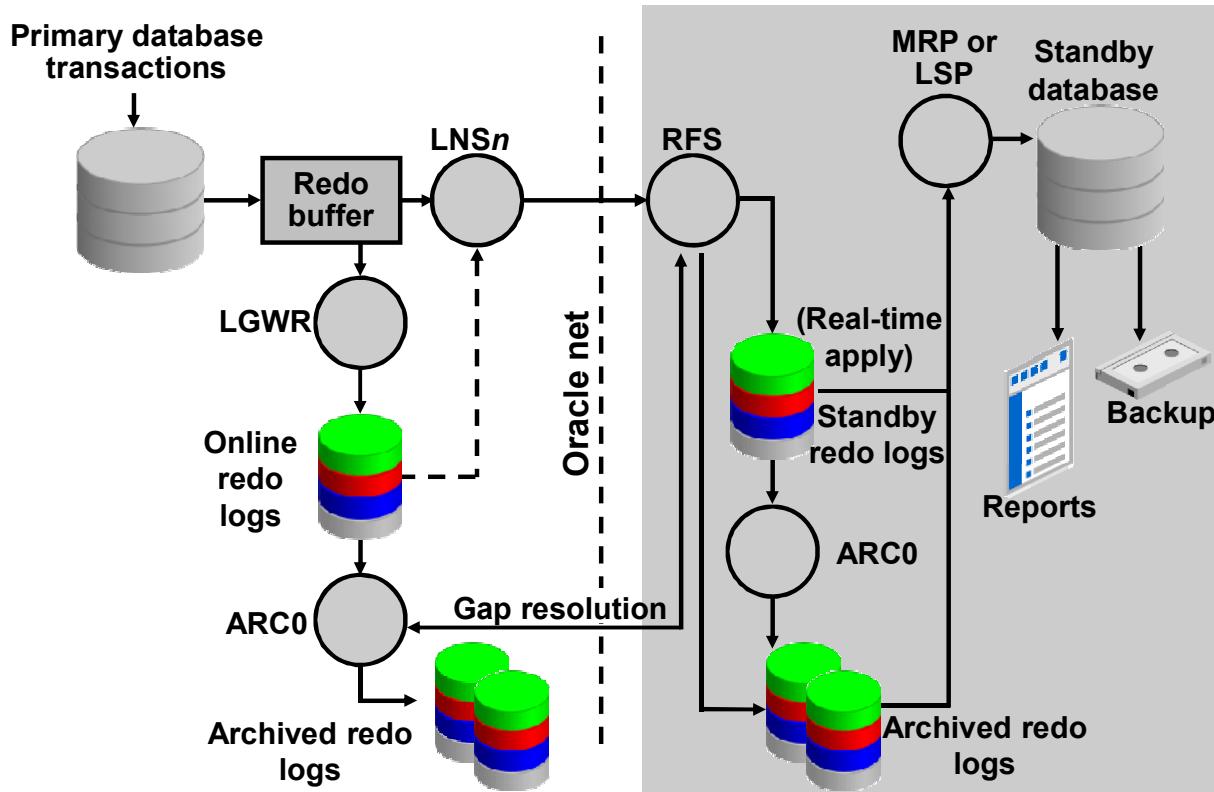
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Data Guard leverages the existing database redo-generation architecture to keep the standby databases in the configuration synchronized with the primary database. By using the existing architecture, Oracle Data Guard minimizes its impact on the primary database.

Oracle Data Guard uses several processes to achieve the automation that is necessary for disaster recovery and high availability. Some of these processes existed before the introduction of Data Guard; others were created specifically to support Oracle Data Guard. These processes are discussed in more detail on the next few pages.

Primary Database Processes



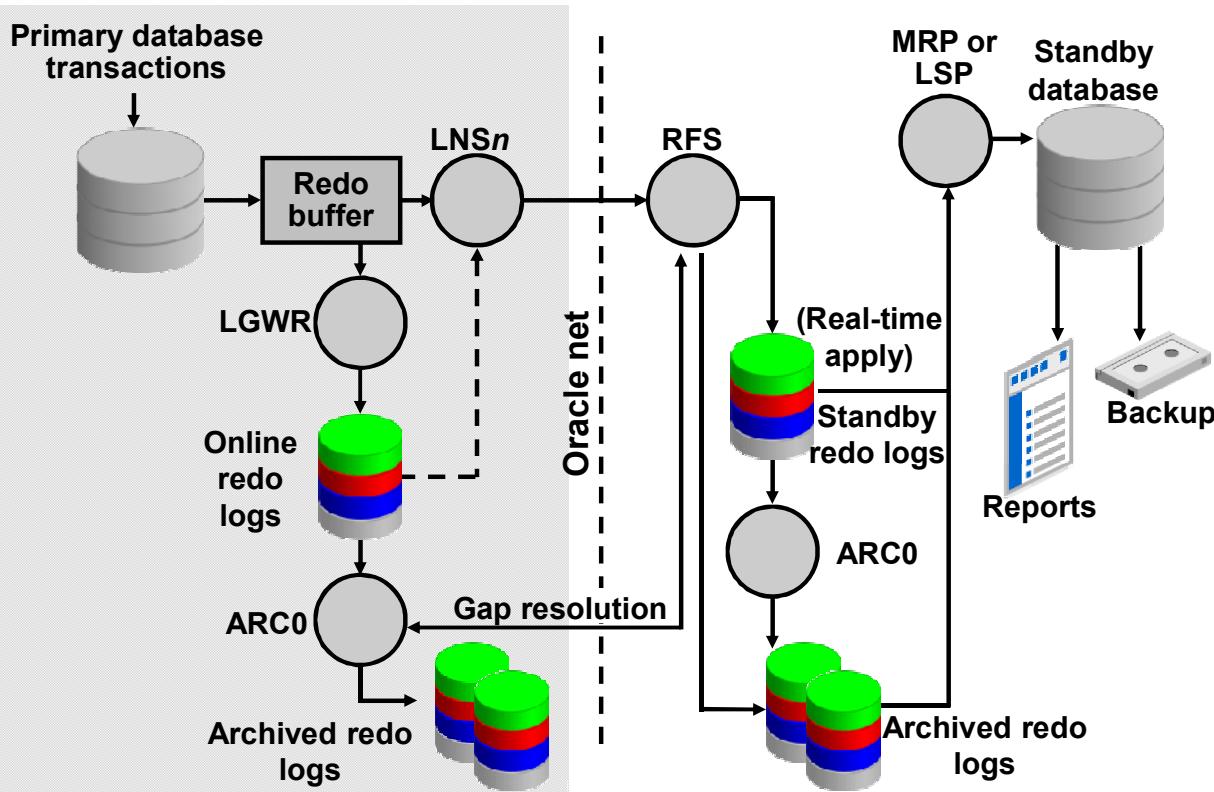
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On the primary database, Data Guard uses the following processes:

- **Log writer (LGWR):** LGWR collects transaction redo information and updates the online redo logs. For each synchronous (SYNC) standby database, LGWR passes the redo to an LNS (Log Writer Network Server) process, which ships the redo directly to the remote file server (RFS) process on the standby database. LGWR waits for confirmation from the LNS process before acknowledging the commit. For asynchronous (ASYNC) standby databases, independent LNS processes read the redo from either the redo log buffer in memory or the online redo log file, and then ship the redo to its standby database. Other than starting the asynchronous LNS processes, LGWR has no interaction with any asynchronous standby destination.
- **Archiver (ARC_n):** The ARC_n process creates a copy of the online redo log files locally for use in a primary database recovery operation. ARC_n is also responsible for shipping redo data to an RFS process at a standby database and for proactively detecting and resolving gaps on all standby databases.

Standby Database Processes



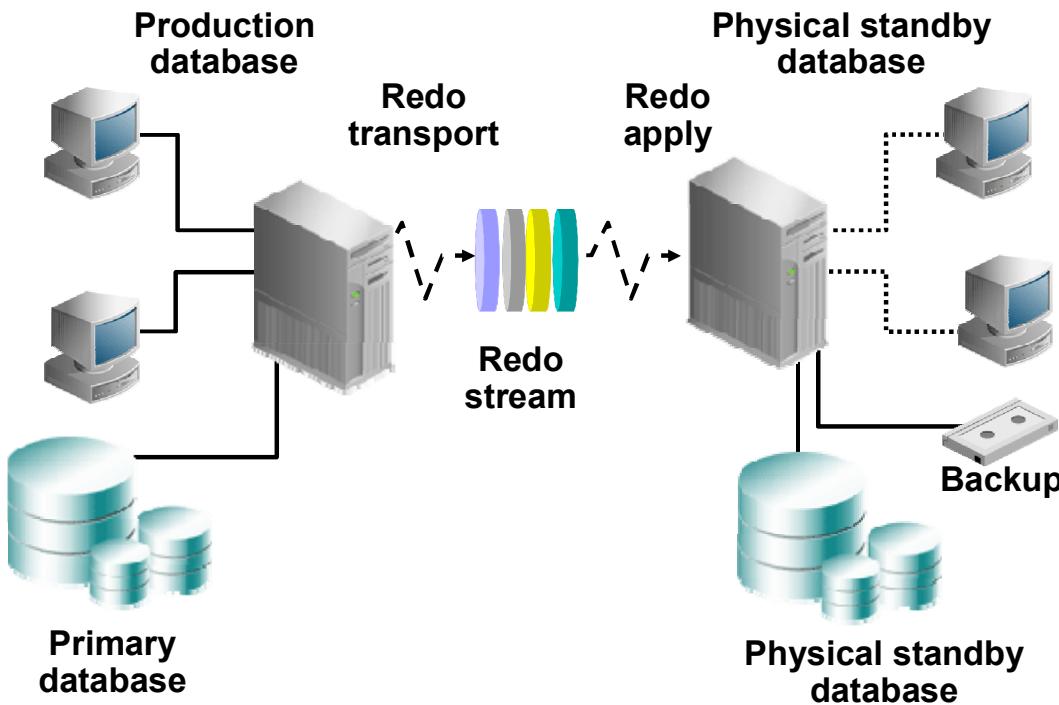
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On the standby database, Data Guard uses the following processes:

- **Remote file server (RFS):** RFS receives redo information from the primary database and can write the redo into standby redo logs or directly to archived redo logs. Each $LNSn$ and $ARCn$ process from the primary database has its own RFS process.
- **Archiver (ARCn):** The $ARCn$ process archives the standby redo logs.
- **Managed recovery (MRP):** For physical standby databases only, MRP applies archived redo log information to the physical standby database. If you start the managed recovery with the `ALTER DATABASE RECOVER MANAGED STANDBY DATABASE` SQL statement, this foreground session performs the recovery. If you use the optional `DISCONNECT [FROM SESSION]` clause, the MRP background process starts. If you use the Data Guard broker to manage your standby databases, the broker always starts the MRP background process for a physical standby database.
- **Logical standby (LSP):** For logical standby databases only, LSP controls the application of archived redo log information to the logical standby database.

Physical Standby Database: Redo Apply Architecture



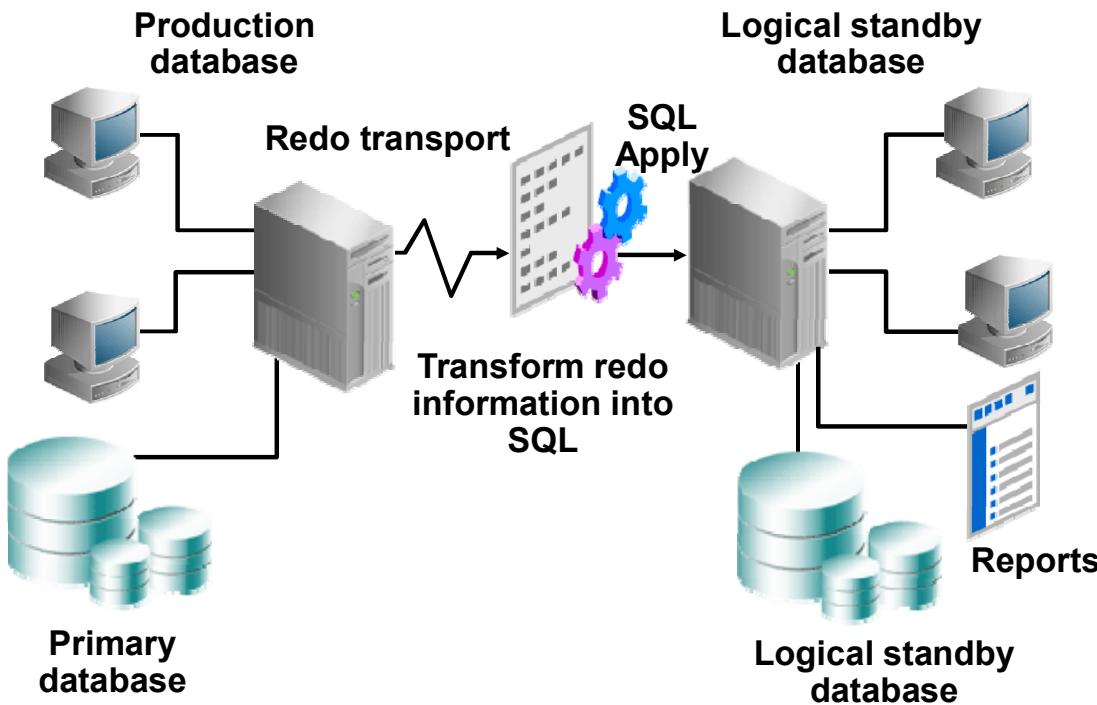
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Data Guard physical standby Redo Apply architecture consists of:

- A production (primary) database, which is linked to one or more standby databases (up to nine) that are identical copies of the production database. The limit of nine standby databases is imposed by the `LOG_ARCHIVE_DEST_n` parameter. In Oracle Database 11g, the maximum number of destinations is 10. One is used as the local archive destination, leaving the other nine for uses such as the standby database.
Note: You can use the Cascaded Redo Log Destinations feature to incorporate more than nine standby databases in your configuration.
- The standby database, which is updated by redo that is automatically shipped from the primary database. The redo can be shipped as it is generated or archived on the primary database. Redo is applied to each standby database by using Oracle media recovery. During planned down time, you can perform a switchover to a standby database. When a failure occurs, you can perform a failover to one of the standby databases. The physical standby database can also be used to back up the primary database.

Logical Standby Database: SQL Apply Architecture



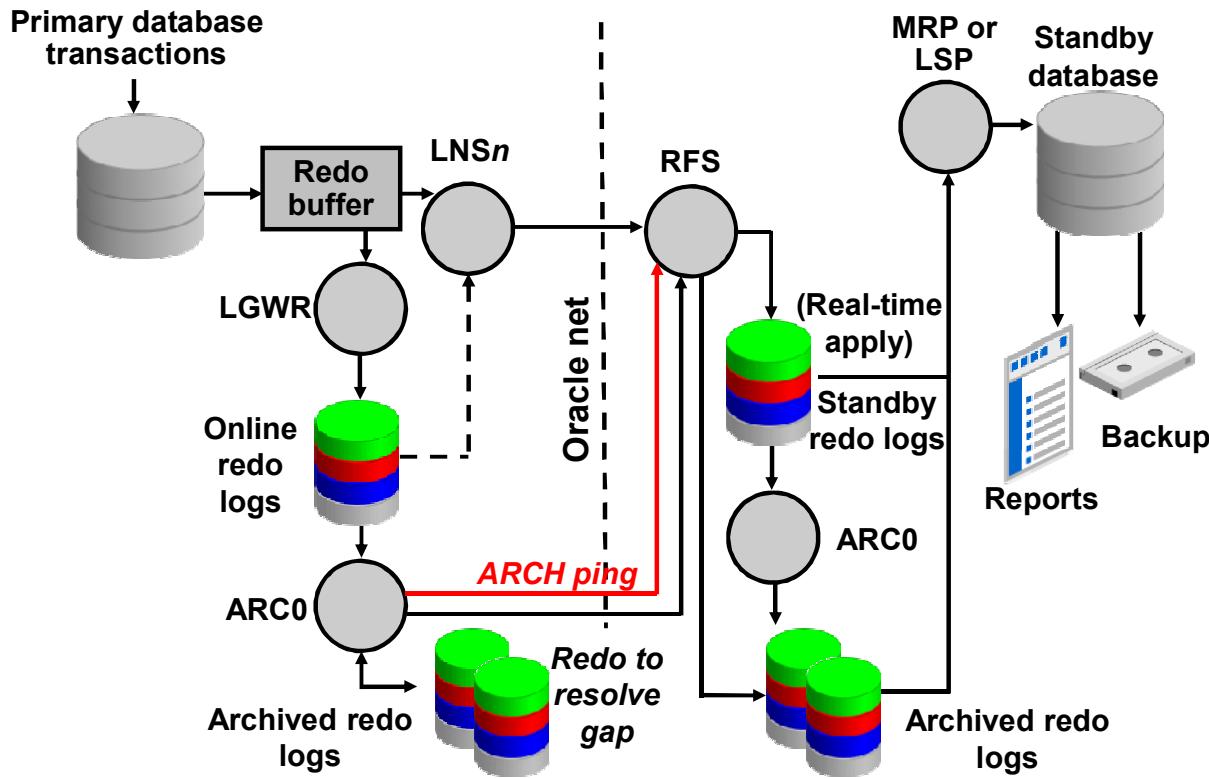
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In a logical standby database configuration, Data Guard SQL Apply uses redo information shipped from the primary system. However, instead of using media recovery to apply changes (as in the physical standby database configuration), the redo data is transformed into equivalent SQL statements by using LogMiner technology. These SQL statements are then applied to the logical standby database. The logical standby database is open in read/write mode and is available for reporting capabilities.

A logical standby database can be used to perform rolling database upgrades, thereby minimizing down time when upgrading to new database patch sets or full database releases.

Automatic Gap Detection and Resolution



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If connectivity is lost between the primary database and one or more standby databases, redo data that is being generated on the primary database cannot be sent to those standby databases. When a connection is reestablished, Data Guard automatically detects that there are missing archived redo log files (referred to as a *gap*), and then automatically transmits the missing archived redo log files to the standby databases by using the *ARCn* processes. The standby databases are synchronized with the primary database without manual intervention by the DBA.

Data Protection Modes

Select the mode to balance cost, availability, performance, and data protection:

- Maximum protection
- Maximum availability
- Maximum performance



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Data Guard provides three high-level modes of data protection that you can configure to balance cost, availability, performance, and transaction protection. You can configure the Data Guard environment to maximize data protection, availability, or performance.

Maximum Protection

This protection mode guarantees that no data loss occurs if the primary database fails. For this level of protection, the redo data that is needed to recover each transaction must be written to both the local online redo log and the standby redo log (used to store redo data received from another database) on at least one standby database before the transaction commits. To ensure that data loss does not occur, the primary database shuts down if a fault prevents it from writing its redo stream to at least one remote standby redo log.

Maximum Availability

This protection mode provides the highest possible level of data protection without compromising the availability of the primary database. As with maximum protection mode, a transaction does not commit until the redo needed to recover that transaction is written to the local online redo log and to at least one remote standby redo log. Unlike maximum protection mode, the primary database does not shut down if a fault prevents it from writing its redo stream to a remote standby redo log. Instead, the primary database operates in an unsynchronized mode until the fault is corrected and all the gaps in the redo log files are resolved. When all the gaps are resolved and the primary database is synchronized with the standby database, the primary database automatically resumes operating in maximum availability mode.

This mode guarantees that no data loss occurs if the primary database fails, but only if a second fault does not prevent a complete set of redo data from being sent from the primary database to at least one standby database.

Maximum Performance (Default)

The default protection mode provides the highest possible level of data protection without affecting the performance of the primary database. This is accomplished by allowing a transaction to commit as soon as the redo data needed to recover that transaction is written to the local online redo log. The primary database's redo data stream is also written to all ASYNC standby databases and is written asynchronously with respect to the commitment of the transactions that create the redo data.

Data Guard Operational Requirements: Hardware and Operating System

Primary database systems and standby database systems may have different:

- CPU architectures
- Operating systems
- Operating system binaries (32-bit or 64-bit)
- Oracle Database binaries (32-bit or 64-bit)



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

These are the requirements for Data Guard:

- The hardware for the primary and standby database systems can be different. For example, the number of CPUs, the memory size, and the storage configuration can differ.
- The operating systems for both databases and operating system binaries can be different. If the primary and standby databases are both on the same server, you must ensure that the operating system enables you to mount two databases with the same name on the same system simultaneously. Certain parameters must be specified to support this configuration, as discussed in the lesson titled “Creating a Physical Standby Database by Using SQL and RMAN Commands.”

Refer to Oracle MetaLink Note 413484.1 for additional information.

Data Guard Operational Requirements: Oracle Database Software

- The same release of Oracle Database Enterprise Edition must be installed for all databases except when you perform a rolling database upgrade by using a logical standby database.
- If any database uses ASM or OMF, all databases should use the same combination.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

- You must install the same release of Oracle Database Enterprise Edition for the primary database and all standby databases in your Data Guard configuration. Oracle Data Guard is available only as a feature of Oracle Database Enterprise Edition; it is not available with Oracle Database Standard Edition.

Note: See the course titled “Oracle Data Guard Concepts and Administration” for information about simulating a standby database environment when using Oracle Database Standard Edition.

- If you use Oracle Automatic Storage Management (ASM) and Oracle Managed Files (OMF) in a Data Guard configuration, you should use ASM and OMF symmetrically on the primary and standby database. If any database in your Data Guard configuration uses ASM, OMF, or both, then every database in the configuration should use the same combination (that is, ASM, OMF, or both).

Note: An exception to this guideline is if you are using Data Guard as a technique to migrate to ASM and/or OMF.

Benefits of Implementing Oracle Data Guard

Oracle Data Guard provides the following benefits:

- Continuous service during disasters or crippling data failures
- Complete data protection against corruption and data loss
- Elimination of idle standby systems
- Flexible configuration of your system to meet requirements for business protection and recovery
- Centralized management



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

- **Continuous service:** With the use of switchover and failover between systems, your business does not need to halt because of a disaster at one location.
- **Complete data protection:** Data Guard guarantees that there is no data loss and provides a safeguard against data corruption and user errors. Redo data is validated when applied to the standby database.
- **Elimination of idle standby systems:** Standby databases can be used for reporting and ad hoc queries in addition to providing a safeguard for disaster recovery. You can also use the physical standby database to off-load backups of the primary database.
- **Flexible configurations:** You can use Data Guard to configure the system to your needs by using the protection modes and several tunable parameters.
- **Centralized management:** You can use Enterprise Manager Grid Control to manage all configurations in your enterprise.

Summary

In this lesson, you should have learned how to:

- Describe the basic components of Oracle Data Guard
- Explain the differences between physical and logical standby databases
- Explain the benefits of creating a Data Guard environment



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

11

Creating a Physical Standby Database by Using SQL and RMAN Commands

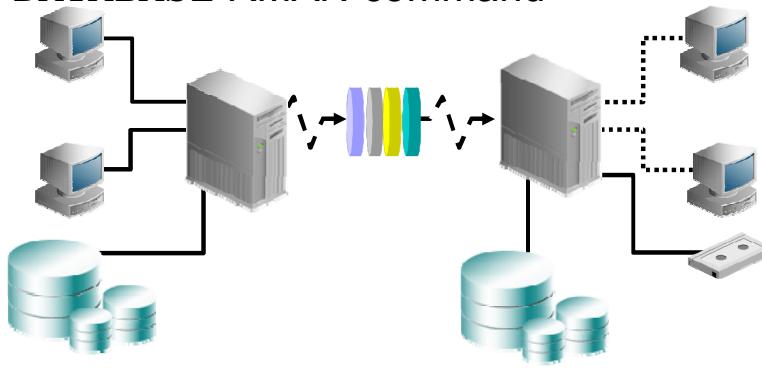


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

- Configure the primary database and Oracle Net Services to support the creation of the physical standby database and role transition
- Create a physical standby database by using the DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE RMAN command



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Steps to Create a Physical Standby Database

1. Prepare the primary database.
2. Set parameters on the physical standby database.
3. Configure Oracle Net Services.
4. Start the standby database instance.
5. Execute the **DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE RMAN** command.
6. Start the transport and application of redo.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You perform the steps listed in the slide to use SQL and RMAN commands to create a physical standby database. Detailed information about each step is provided in the remaining slides of the lesson.

Note: See *Oracle Data Guard Concepts and Administration* for detailed information about creating a physical standby database by using only SQL commands.

Preparing the Primary Database

- Enable FORCE LOGGING at the database level.
- Create a password file if required.
- Create standby redo logs.
- Set initialization parameters.
- Enable archiving.

```
SQL> SHUTDOWN IMMEDIATE;
SQL> STARTUP MOUNT;
SQL> ALTER DATABASE ARCHIVELOG;
SQL> ALTER DATABASE OPEN;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The FORCE LOGGING mode determines whether the Oracle database server logs all changes in the database (except for changes to temporary tablespaces and temporary segments).

Note: Additional information about enabling FORCE LOGGING follows in this lesson.

Unless you have configured Oracle Advanced Security and public key infrastructure (PKI) certificates, every database in a Data Guard configuration must use a password file, and the password for the `SYS` user must be identical on every system for redo data transmission to succeed. For details about creating a password file, see the *Oracle Database Administrator's Guide*.

A standby redo log is used to store redo received from another Oracle database.

Note: Additional information about creating standby redo log files is provided in this lesson.

On the primary database, you define initialization parameters that control redo transport services while the database is in the primary role. There are other parameters that you need to add that control the receipt of the redo data and log apply services when the primary database is transitioned to the standby role. Additional information about setting initialization parameters is provided in this lesson.

Note: The Data Guard broker requires the use of a server parameter file.

If archiving is not enabled, issue the `ALTER DATABASE ARCHIVELOG` command to put the primary database in ARCHIVELOG mode and enable automatic archiving. See the *Oracle Database Administrator's Guide* for additional information about archiving.

FORCE LOGGING Mode

- FORCE LOGGING mode is recommended to ensure data consistency.
- FORCE LOGGING forces redo to be generated even when NOLOGGING operations are executed.
- Temporary tablespaces and temporary segments are not logged.
- FORCE LOGGING is recommended for both physical and logical standby databases.
- Issue the following command on the primary database:

```
SQL> ALTER DATABASE FORCE LOGGING;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

FORCE LOGGING mode determines whether the Oracle database server logs all changes in the database (except for changes to temporary tablespaces and temporary segments). The [NO] FORCE LOGGING clause of the ALTER DATABASE command contains the following settings:

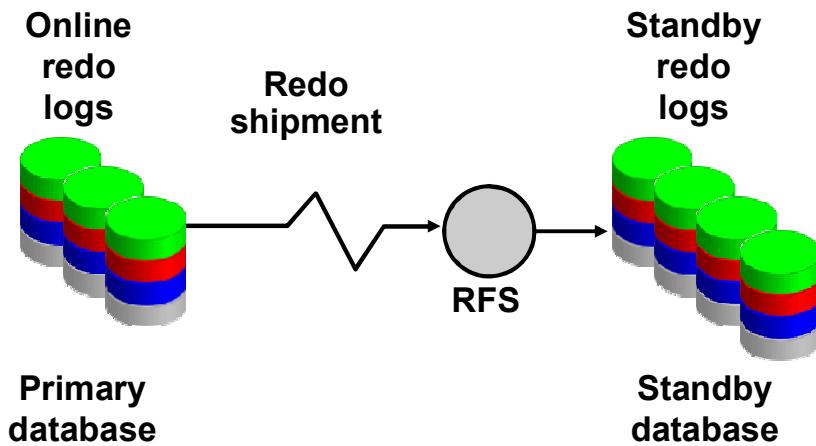
- **FORCE LOGGING:** This setting takes precedence over (and is independent of) any NOLOGGING or FORCE LOGGING settings that you specify for individual tablespaces and any NOLOGGING setting that you specify for individual database objects. All ongoing, unlogged operations must finish before forced logging can begin.
- **NOFORCE LOGGING:** Places the database in NOFORCE LOGGING mode. This is the default.

The FORCE_LOGGING column in V\$DATABASE contains a value of YES if the database is in FORCE LOGGING mode.

Although the database can be placed in FORCE LOGGING mode when the database is OPEN, the mode does not change until the completion of all operations that are currently running in NOLOGGING mode. Therefore, it is recommended that you enable FORCE LOGGING mode when the database is in the MOUNT state.

Note: You should enable FORCE LOGGING before performing the backup operation to create the standby database, and then maintain FORCE LOGGING mode for as long as the Data Guard configuration exists.

Configuring Standby Redo Logs



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A standby redo log is used only when the database is in the standby role to store redo data received from the primary database. Standby redo logs form a separate pool of log file groups.

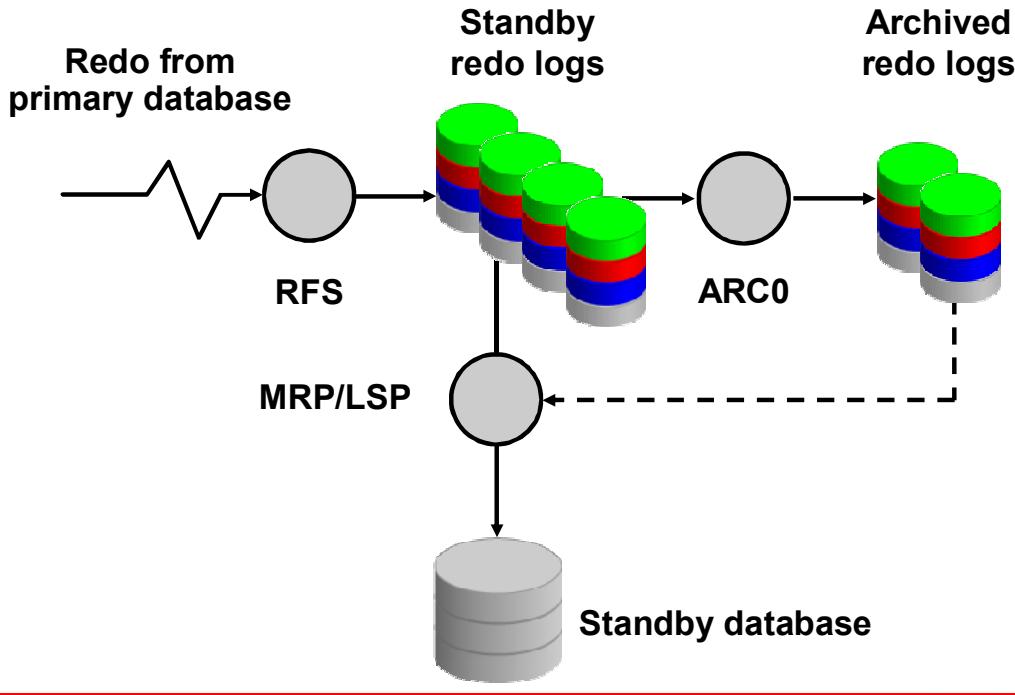
Configuring standby redo log files is highly recommended for all databases in a Data Guard configuration to aid in role reversal.

A standby redo log is required to implement:

- Real-time apply
- Cascaded redo log destinations

Note: By configuring the standby redo log on the primary database, the standby redo log is created automatically on the standby database when you execute the DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE RMAN command.

Creating Standby Redo Logs



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You must create at least the same number of standby redo log file groups as there are online redo log file groups on the primary database. It is highly recommended that you have one more standby redo log group than you have online redo log groups as the primary database. In addition, the files should be the same size as the primary database's online redo logs. If your online redo log files are of different sizes, the remote file server (RFS) process automatically uses the same size standby redo log as the online redo log file.

The RFS process writes to an archive redo log file if any of the following conditions are met:

- There are no standby redo logs.
- It cannot find a standby redo log that is the same size as the incoming online redo log file.
- All standby redo logs of the correct size have not yet been archived.

Using SQL to Create Standby Redo Logs

Create standby redo logs on the primary database:

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE
  2  '/u01/app/oracle/oradata/orcl/srl01.log'
  3  SIZE 50M;
Database altered.

  4  ALTER DATABASE ADD STANDBY LOGFILE
  5  '/u01/app/oracle/oradata/orcl/srl02.log'
  6  SIZE 50M;
Database altered.
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can create standby redo logs by using the ADD STANDBY LOGFILE clause of the ALTER DATABASE statement. Although standby redo logs are used only when the database is operating in the standby role, you should create standby redo logs on the primary database so that switching roles does not require additional DBA intervention.

You should create standby redo log files on the primary database prior to using the DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE RMAN command so that RMAN creates standby redo log files automatically on the standby database.

Create standby redo log file groups by using the following guidelines:

- Each standby redo log file must be at least as large as the largest redo log file in the redo source database. For administrative ease, Oracle recommends that all redo log files in the redo source database and the redo transport destination be of the same size.
- The standby redo log should have at least one more redo log group than the redo log on the redo source database.

Viewing Standby Redo Log Information

View information about the standby redo logs:

```
SQL> SELECT group#, type, member FROM v$logfile
  2 WHERE type = 'STANDBY';
GROUP# TYPE      MEMBER
-----
4 STANDBY /u01/app/oracle/oradata/pc00prmy/srl01.log
5 STANDBY /u01/app/oracle/oradata/pc00prmy/srl02.log
6 STANDBY /u01/app/oracle/oradata/pc00prmy/srl03.log
7 STANDBY /u01/app/oracle/oradata/pc00prmy/srl04.log

SQL> SELECT group#, dbid, thread#, sequence#, status
  2 FROM v$standby_log;
GROUP# DBID          THREAD#  SEQUENCE# STATUS
-----
4 UNASSIGNED          0          0 UNASSIGNED
5 UNASSIGNED          0          0 UNASSIGNED
6 UNASSIGNED          0          0 UNASSIGNED
7 UNASSIGNED          0          0 UNASSIGNED
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To verify that standby redo logs were created, query V\$STANDBY_LOG or V\$LOGFILE.

Setting Initialization Parameters on the Primary Database to Control Redo Transport

Parameter Name	Description
LOG_ARCHIVE_CONFIG	Specifies the unique database name for each database in the configuration
LOG_ARCHIVE_DEST_n	Controls redo transport services
LOG_ARCHIVE_DEST_STATE_n	Specifies the destination state
ARCHIVE_LAG_TARGET	Forces a log switch after the specified number of seconds
LOG_ARCHIVE_TRACE	Controls output generated by the archiver process



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Setting Initialization Parameters on the Primary Database

On the primary database, you define initialization parameters that control redo transport services while the database is in the primary role. These parameters are described in more detail in the following slides.

Setting LOG_ARCHIVE_CONFIG

Specify the DG_CONFIG attribute to list the DB_UNIQUE_NAME for the primary database and each standby database in the Data Guard configuration.

```
LOG_ARCHIVE_CONFIG='DG_CONFIG=(pc00prmy,pc00sby1)'
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Specify the DG_CONFIG attribute of the LOG_ARCHIVE_CONFIG parameter to list the DB_UNIQUE_NAME of the primary and standby databases in the Data Guard configuration. By default, the LOG_ARCHIVE_CONFIG parameter enables the database to send and receive redo.

Use the V\$DATAGUARD_CONFIG view to see the unique database names defined with the DB_UNIQUE_NAME and LOG_ARCHIVE_CONFIG initialization parameters; you can thus view the Data Guard environment from any database in the configuration. The first row of the view lists the unique database name of the current database that was specified with the DB_UNIQUE_NAME initialization parameter. Additional rows reflect the unique database names of the other databases in the configuration that were specified with the DG_CONFIG keyword of the LOG_ARCHIVE_CONFIG initialization parameter.

The following example illustrates the use of V\$DATAGUARD_CONFIG:

```
SQL> show parameter log_archive_config

NAME          TYPE        VALUE
-----
log_archive_config    string    dg_config=(pc00prmy,pc00sby1)

SQL> SELECT * FROM v$dataguard_config;

DB_UNIQUE_NAME
-----
pc00prmy
pc00sby1
```

Setting LOG_ARCHIVE_DEST_n

- Specify LOG_ARCHIVE_DEST_n parameters for:
 - Local archiving (if not using the flash recovery area)
 - Standby database location
- Include (at a minimum) one of the following:
 - LOCATION: Specifies a valid path name
 - SERVICE: Specifies a valid Oracle Net Services name referencing a standby database
- Include a LOG_ARCHIVE_DEST_STATE_n parameter for each defined destination.

```
LOG_ARCHIVE_DEST_1=
'SERVICE=pc00sby1
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE)
DB_UNIQUE_NAME=pc00sby1'
LOG_ARCHIVE_DEST_STATE_1=ENABLE
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

By using the various LOG_ARCHIVE_DEST_n attributes, you define most of the settings for the Data Guard configuration. The Redo Transport Service is directly controlled by these settings. There are a number of different attributes that can be set for each LOG_ARCHIVE_DEST_n parameter. Most have defaults that are adequate for most configurations. See *Oracle Data Guard Concepts and Administration* for a complete list and a description of each.

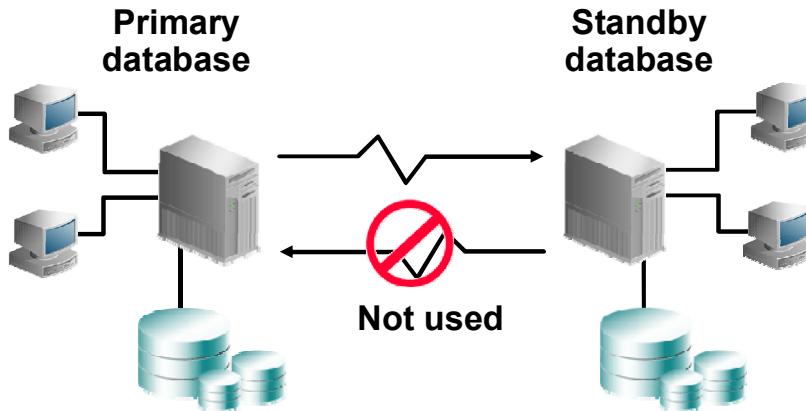
You should specify a LOG_ARCHIVE_DEST_n parameter (where n is an integer from 1 to 10) for the local archiving destination and one for the standby location. If you are using the flash recovery area for local archiving, LOG_ARCHIVE_DEST_10 is set automatically to

USE_DB_RECOVERY_FILE_DEST. Query the V\$ARCHIVE_DEST view to see current settings of the LOG_ARCHIVE_DEST_n initialization parameter.

All defined LOG_ARCHIVE_DEST_n parameters must contain, at a minimum, either a LOCATION attribute or a SERVICE attribute.

In addition, you must have a LOG_ARCHIVE_DEST_STATE_n parameter for each defined destination. LOG_ARCHIVE_DEST_STATE_n defaults to ENABLE.

Specifying Role-Based Destinations



```
log_archive_dest_1 =
'service=pc00sby1 async
 valid_for=
 (online_logfile,
 primary_role)
db_unique_name=pc00sby1'
```

```
log_archive_dest_1 =
'service=pc00prmy async
 valid_for=
 (online_logfile,
 primary_role)
db_unique_name=pc00prmy'
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The **VALID_FOR** attribute of the **LOG_ARCHIVE_DEST_n** initialization parameter enables you to identify exactly when the archive destination is to be used, as well as which type of log file it is used for. The attribute uses a keyword pair to identify the redo log type as well as the database role. Using this attribute enables you to set up parameters in anticipation of switchover and failover operations.

In the example in the slide, there is a destination on the standby database and the primary database defined with the **VALID_FOR** setting shown. This destination is to be used on the standby database only after a switchover, when the standby becomes a primary. The destination on the old primary is ignored when it becomes a standby.

You supply two values for the **VALID_FOR** attribute: **redo_log_type** and **database_role**.

The **redo_log_type** keywords are:

- **ONLINE_LOGFILE**: This destination is used only when archiving online redo log files.
- **STANDBY_LOGFILE**: This destination is used only when archiving standby redo log files or receiving archive logs from another database.
- **ALL_LOGFILES**: This destination is used when archiving either online or standby redo log files.

The *database_role* keywords are the following:

- **PRIMARY_ROLE**: This destination is used only when the database is in the primary database role.
- **STANDBY_ROLE**: This destination is used only when the database is in the standby (logical or physical) role.
- **ALL_ROLES**: This destination is used when the database is in either the primary or the standby (logical or physical) role.

Note: Because the keywords are unique, the *archival_source* and *database_role* values can be specified in any order.

For example, `VALID_FOR= (PRIMARY_ROLE, ONLINE_LOGFILE)` is functionally equivalent to `VALID_FOR= (ONLINE_LOGFILE, PRIMARY_ROLE)`.

Combinations for VALID_FOR

Combination	Primary	Physical	Logical
ONLINE_LOGFILE, PRIMARY_ROLE	Valid	Ignored	Ignored
ONLINE_LOGFILE, STANDBY_ROLE	Ignored	Ignored	Valid
ONLINE_LOGFILE, ALL_ROLES	Valid	Ignored	Valid
STANDBY_LOGFILE, STANDBY_ROLE	Ignored	Valid	Valid
STANDBY_LOGFILE, ALL_ROLES	Ignored	Valid	Valid
ALL_LOGFILES, PRIMARY_ROLE	Valid	Ignored	Ignored
ALL_LOGFILES, STANDBY_ROLE	Ignored	Valid	Valid
ALL_LOGFILES, ALL_ROLES	Valid	Valid	Valid

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the table in the slide, *Valid* indicates that the archive log destination is used in a database that is in the role defined by the column heading. *Ignored* means that the archive log destination is not appropriate and that a destination of this type is ignored. An ignored destination does not generate an error.

There is only one invalid combination: STANDBY_LOGFILE, PRIMARY_ROLE. Specifying this combination causes an error for all database roles. If it is set, you receive the following error at startup:

ORA-16026: The parameter LOG_ARCHIVE_DEST_n contains an invalid attribute value

Note: Both single and plural forms of the keywords are valid. For example, you can specify either PRIMARY_ROLE or PRIMARY_ROLES, as well as ONLINE_LOGFILE or ONLINE_LOGFILES.

Defining the Redo Transport Mode

Use the attributes of `LOG_ARCHIVE_DEST_n`:

- **SYNC** and **ASYNC**
 - Specify that network I/O operations are to be performed synchronously or asynchronously when using LGWR.
 - **ASYNC** is the default.
- **AFFIRM** and **NOAFFIRM**
 - Ensure that redo was successfully written to disk on the standby destination.
 - **NOAFFIRM** is the default when **ASYNC** is specified; **AFFIRM** is the default when **SYNC** is specified.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following attributes of the `LOG_ARCHIVE_DEST_n` initialization parameter define the redo transport mode that is used by the primary database to send redo to the standby database.

- **SYNC**: Specifies that redo data generated by a transaction must have been received at a destination that has this attribute before the transaction can commit; otherwise, the destination is deemed to have failed. In a configuration with multiple `SYNC` destinations, the redo must be processed as described here for every `SYNC` destination.
- **ASYNC (default)**: Specifies that redo data generated by a transaction need not have been received at a destination that has this attribute before the transaction can commit
- **AFFIRM**: Specifies that a redo transport destination acknowledges received redo data after writing it to the standby redo log
- **NOAFFIRM**: Specifies that a redo transport destination acknowledges received redo data before writing it to the standby redo log

If neither the `AFFIRM` nor the `NOAFFIRM` attribute is specified, the default is `AFFIRM` when the `SYNC` attribute is specified and `NOAFFIRM` when the `ASYNC` attribute is specified.

Setting Initialization Parameters on the Primary Database

- Specify parameters when standby databases have disk or directory structures that differ from the primary database.
- Use parameters when the primary database is transitioned to a standby database.

Parameter Name	Description
DB_FILE_NAME_CONVERT	Converts primary database file names
LOG_FILE_NAME_CONVERT	Converts primary database log file names
STANDBY_FILE_MANAGEMENT	Controls automatic standby file management



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The parameters listed in the slide are required if the disk configuration is not the same for the primary and standby databases. The parameters are also applicable when the primary database is transitioned to a standby database.

Specifying Values for DB_FILE_NAME_CONVERT

- DB_FILE_NAME_CONVERT must be defined on standby databases that have different disk or directory structures from the primary.
- Multiple pairs of file names can be listed in the DB_FILE_NAME_CONVERT parameter.
- DB_FILE_NAME_CONVERT applies only to a physical standby database.
- DB_FILE_NAME_CONVERT can be set in the DUPLICATE RMAN script.

```
DB_FILE_NAME_CONVERT = ('/oracle1/dba/',
                        '/ora1/stby_dba/',
                        '/oracle2/dba/',
                        '/ora2/stby_dba/')
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When files are added to the standby database, the DB_FILE_NAME_CONVERT parameter is used to convert the data file name on the primary database to a data file name on the standby database. The file must exist and be writable on the physical standby database; if it is not, the recovery process halts with an error.

You specify the path name and file name location of the primary database data files followed by the standby location by setting the value of this parameter to two strings. The first string is the pattern found in the data file names on the primary database. The second string is the pattern found in the data file names on the physical standby database. You can use as many pairs of primary and standby replacement strings as required. You can use single or double quotation marks. Parentheses are optional.

In the example in the slide, /oracle1/dba/ and /oracle2/dba/ are used to match file names coming from the primary database. /ora1/stby_dba/ and /ora2/stby_dba/ are the corresponding strings for the physical standby database. A file on the primary database named /oracle1/dba/system01.dbf is converted to /ora1/stby_dba/system01.dbf on the standby database.

Note: If the standby database uses Oracle Managed Files (OMF), do not set the DB_FILE_NAME_CONVERT parameter.

Specifying Values for LOG_FILE_NAME_CONVERT

- LOG_FILE_NAME_CONVERT is similar to DB_FILE_NAME_CONVERT.
- LOG_FILE_NAME_CONVERT must be defined on standby databases that have different disk or directory structures from the primary.
- LOG_FILE_NAME_CONVERT applies only to a physical standby database.
- LOG_FILE_NAME_CONVERT can be set in the DUPLICATE RMAN script.

```
LOG_FILE_NAME_CONVERT = ('/oracle1/logs/',
                          '/ora1/stby_logs/')
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The LOG_FILE_NAME_CONVERT parameter is used to convert the name of a redo log file on the primary database to the name of a redo log file on the standby database. Adding a redo log file to the primary database requires adding a corresponding file to the standby database. When the standby database is updated, this parameter is used to convert the log file name from the primary database to the log file name on the standby database. This parameter is required if the standby database is on the same system as the primary database or on a separate system that uses different path names.

Specify the location of the primary database online redo log files followed by the standby location. The use of parentheses is optional.

Note: If the standby database uses OMF, do not set the LOG_FILE_NAME_CONVERT parameter.

Specifying a Value for **STANDBY_FILE_MANAGEMENT**

- STANDBY_FILE_MANAGEMENT is used to maintain consistency when you add or delete a data file on the primary database.
 - MANUAL (default)
 - Data files must be manually added to the standby database.
 - AUTO
 - Adds the data file automatically to the standby database.
 - Certain ALTER statements are no longer allowed on the standby database.
- STANDBY_FILE_MANAGEMENT applies to the primary database and physical standby database.

STANDBY_FILE_MANAGEMENT = auto



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When STANDBY_FILE_MANAGEMENT is set to AUTO, you cannot execute the following commands on the standby database:

- ALTER DATABASE RENAME
- ALTER DATABASE ADD/DROP LOGFILE [MEMBER]
- ALTER DATABASE ADD/DROP STANDBY LOGFILE MEMBER
- ALTER DATABASE CREATE DATAFILE AS . . .

When you add a log file to the primary database and want to add it to the physical standby database as well (or when you drop a log file from the primary and want to drop it from the physical), you must do the following:

1. Set STANDBY_FILE_MANAGEMENT to MANUAL on the physical standby database.
2. Add the redo log files to (or drop them from) the primary database.
3. Add them to (or drop them from) the standby database.
4. Reset to AUTO afterward on the standby database.

Example: Setting Initialization Parameters on the Primary Database

```
DB_NAME=pc00prmy
DB_UNIQUE_NAME=pc00prmy
LOG_ARCHIVE_CONFIG='DG_CONFIG=(pc00prmy,pc00sby1)'
CONTROL_FILES='/u01/app/oracle/oradata/pc00prmy/control1.ctl',
'/u01/app/oracle/oradata/pc00prmy/control2.ctl'
LOG_ARCHIVE_DEST_1=
'SERVICE=pc00sby1
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE)
DB_UNIQUE_NAME=pc00sby1'
LOG_ARCHIVE_DEST_STATE_1=ENABLE
REMOTE_LOGIN_PASSWORDFILE=EXCLUSIVE
LOG_ARCHIVE_FORMAT=%t %s %r.arc
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example in the slide, assume that the primary database is named pc00prmy and the standby is named pc00sby1. For each, there is an Oracle Net Services name defined.

There are additional parameters you need to add that control the receipt of the redo data and log apply services when the primary database is transitioned to the standby role:

```
DB_FILE_NAME_CONVERT='/u01/app/oracle/oradata/pc00sby1/',
'/u01/app/oracle/oradata/pc00prmy/'
LOG_FILE_NAME_CONVERT='/u01/app/oracle/oradata/pc00sby1/',
'/u01/app/oracle/oradata/pc00prmy/'
STANDBY_FILE_MANAGEMENT=AUTO
```

Specifying these initialization parameters configures the primary database to resolve gaps, converts new data file and log file path names from a new primary database, and archives the incoming redo data when this database is in the standby role.

Creating an Oracle Net Service Name for Your Physical Standby Database

Use Oracle Net Manager to update the `tnsnames.ora` file:

```
PC00SBY1 =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP)
        (HOST = edt3r17p1.us.oracle.com)
        (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = pc00sby1.us.oracle.com)
    )
  )
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use Oracle Net Manager to define a network service name for your physical standby database. The slide shows the entry in the `tnsnames.ora` file that was generated by Oracle Net Manager.

Note: This entry is used to connect to the standby database when invoking RMAN and executing the DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE command.

Creating an Entry for Your Standby Database for the Listener

Use Oracle Net Manager to configure an entry for your standby database in the `listener.ora` file:

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = pc00sby1.us.oracle.com)
      (ORACLE_HOME =
/u01/app/oracle/product/11.1.0/db_1)
      (SID_NAME = pc00sby1)
    )
  )
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use Oracle Net Manager to configure a new listener (if necessary) or to update the `listener.ora` file with an entry for your physical standby database. The slide shows the entry in the `listener.ora` file that was generated by Oracle Net Manager.

Note: This entry is needed because we start the instance in NOMOUNT mode.

Copying Your Primary Database Password File to the Physical Standby Database Host

1. Copy the primary database password file to the \$ORACLE_HOME/dbs directory on the standby database host.
2. Rename the file for your standby database: orapw<SID>.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You must create a password file for your physical standby database by copying the primary database password file to the physical standby database host and renaming it.

Creating an Initialization Parameter File for the Physical Standby Database

Create an initialization parameter file containing a single parameter:

```
DB_NAME=pc00sby1
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Create a text initialization parameter file containing only the `DB_NAME` initialization parameter. This initialization parameter file is used to start the physical standby database in `NOMOUNT` mode prior to the execution of the `DUPLEX TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE RMAN` command. When you execute this command, RMAN creates a server parameter file for the standby database.

Creating Directories for the Physical Standby Database

1. Create the audit trail directory in \$ORACLE_BASE/admin:

```
[oracle@edt3r17p1-orcl ~] $ cd  
/u01/app/oracle/admin  
[oracle@edt3r17p1-orcl admin] $ ls  
orcl  
[oracle@edt3r17p1-orcl admin] $ mkdir pc00sby1  
[oracle@edt3r17p1-orcl admin] $ cd pc00sby1  
[oracle@edt3r17p1-orcl orclsby1] $ mkdir adump
```

2. Create a directory for the data files in the \$ORACLE_BASE/oradata directory:

```
[oracle@edt3r17p1-orcl oradata] $ mkdir pc00sby1  
[oracle@edt3r17p1-orcl oradata] $ ls  
orcl pc00sby1
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Create a directory for the physical standby database in the \$ORACLE_BASE/admin directory.

Create the audit trail directory named adump under the database directory in \$ORACLE_BASE/admin.

Create a directory for the physical standby database data files in the \$ORACLE_BASE/oradata directory.

Starting the Physical Standby Database

Start the physical standby database in NOMOUNT mode:

```
SQL> startup nomount pfile=$HOME/dbs/pc00sby1.ora
ORACLE instance started.

Total System Global Area  150667264 bytes
Fixed Size                  1298472 bytes
Variable Size                92278744 bytes
Database Buffers            50331648 bytes
Redo Buffers                 6758400 bytes
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Set the `ORACLE_SID` environment variable to your physical standby database. Start the physical standby database in NOMOUNT mode by using the text initialization parameter file.

Setting FAL_CLIENT and FAL_SERVER Initialization Parameters

- Fetch archive log (FAL):
 - Provides a client/server mechanism for resolving gaps detected in the range of archived redo logs that are generated at the primary database and received at the standby database.
 - Applicable for physical standby databases only.
 - Process is started only when needed, and shuts down as soon as it is finished.
- FAL_CLIENT: Specifies the FAL client name that is used by the FAL service
- FAL_SERVER: Specifies the FAL server for a standby database

```
FAL_CLIENT = 'pc00sby1'  
FAL_SERVER = 'pc00prmy'
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On physical standby databases, fetch archive log (FAL) provides a client/server mechanism for resolving gaps detected in the range of archived redo logs that are generated at the primary database and received at the standby database. The FAL process is started only when needed, and shuts down as soon as it is finished. It is very likely you will not see this process running.

The FAL_CLIENT initialization parameter specifies the FAL client name (Oracle Net service name) that is used by the FAL service, configured through the FAL_SERVER parameter, to refer to the FAL client.

The FAL_SERVER initialization parameter specifies the FAL server (Oracle Net service name) for a standby database.

Creating an RMAN Script to Create the Physical Standby Database

Create an RMAN script to create the physical standby database:

```
run {
    allocate channel prmy1 type disk;
    allocate channel prmy2 type disk;
    allocate channel prmy3 type disk;
    allocate channel prmy4 type disk;
    allocate auxiliary channel stby type disk;

    duplicate target database for standby
        from active database
```

Note: The script continues in the next slide.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Create an RMAN script containing the DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE command.

Note: You can use the CONFIGURE ... PARALLELISM *integer* command to configure automatic channels for the specified device type. For additional information, see the *Oracle Database Backup and Recovery Reference*.

Creating an RMAN Script to Create the Physical Standby Database

```
spfile
parameter_value_convert 'pc00prmy','pc00sby1'
set db_unique_name='pc00sby1'
set db_file_name_convert='/pc00prmy/','/pc00sby1/'
set log_file_name_convert='/pc00prmy/','/pc00sby1/'
set control_files=
  '/u01/app/oracle/oradata/pc00sby1.ctl'
set log_archive_max_processes='5'
set fal_client='pc00sby1'
set fal_server='pc00prmy'
set standby_file_management='AUTO'
set log_archive_config='dg_config=(pc00prmy,pc00sby1)'
set log_archive_dest_1='service=pc00prmy ASYNC
  valid_for=(ONLINE_LOGFILE,PRIMARY_ROLE)
  db_unique_name=pc00prmy';
}
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the RMAN script, specify the settings for the physical standby initialization parameters.

Creating the Physical Standby Database

1. Invoke RMAN and connect to the primary database and the physical standby database.
2. Execute the RMAN script to create the physical standby database.

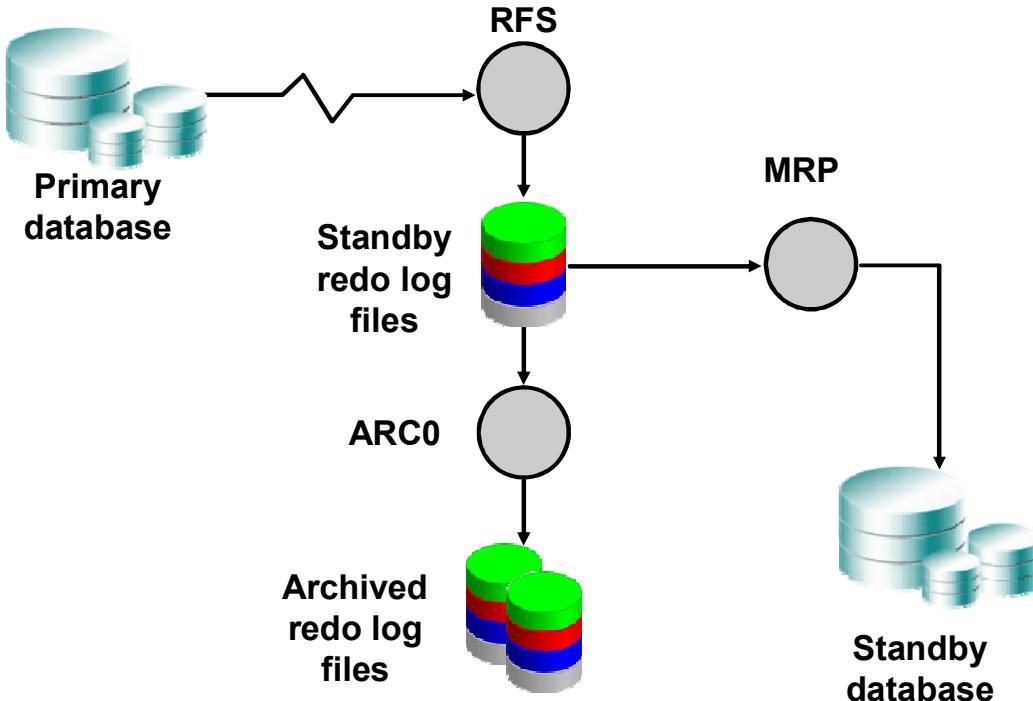
```
RMAN> connect target sys/oracle
RMAN> connect auxiliary sys/oracle@pc00sby1
RMAN> @cr_phys_standby
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Connect to the primary database instance (target) and physical standby database instance (auxiliary). Execute the script that you created.

Enabling Real-Time Apply



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When you enable the optional real-time apply feature, log apply services apply the redo data from standby redo log files in real time (at the same time the log files are being written to) as opposed to recovering redo from archived redo log files when a log switch occurs. If for some reason the apply service is unable to keep up (for example, if you have a physical standby in read-only mode for a period of time), then the apply service automatically goes to the archived redo log files as needed. The apply service also tries to catch up and go back to reading the standby redo log files as soon as possible.

Real-time application of redo information provides a number of benefits, including faster switchover and failover operations, up-to-date results after you change a physical standby database to read-only, up-to-date reporting from a logical standby database, and the ability to leverage larger log files. Having larger log files with real-time apply is desirable because the apply service stays with a log longer and the overhead of switching has less impact on the real-time apply processing.

The RECOVERY_MODE column of the V\$ARCHIVE_DEST_STATUS view contains the value MANAGED REAL TIME APPLY when log apply services are running in real-time apply mode.

If you define a delay on a destination (with the `DELAY` attribute) and use real-time apply, the delay is ignored.

For physical standby databases, the managed recovery process (MRP) applies the redo from the standby redo log files after the remote file server (RFS) process finishes writing. To start real-time apply for a physical standby database, issue the following command:

```
ALTER DATABASE RECOVER MANAGED STANDBY DATABASE  
    USING CURRENT LOGFILE;
```

Note: Standby redo log files are required for real-time apply. It is highly recommended that you have one more standby redo log group than the number of online log groups on the primary database.

Real-time apply is supported and automatically enabled by the broker.

Starting Redo Apply

Execute the following command on the standby database to start Redo Apply:

```
SQL> ALTER DATABASE
  2  RECOVER MANAGED STANDBY DATABASE
  3  USING CURRENT LOGFILE
  4  DISCONNECT FROM SESSION;
```



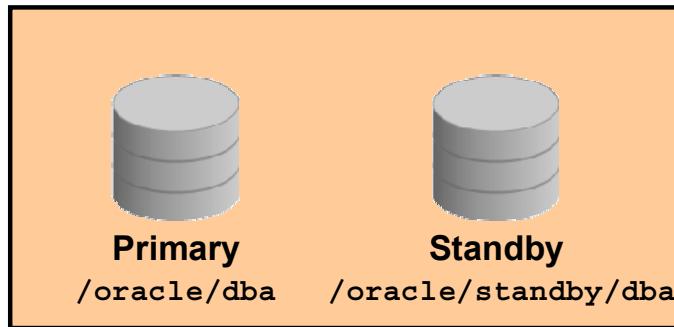
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On the standby database, issue the `ALTER DATABASE RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE` SQL command to start Redo Apply. This statement automatically mounts the database. In addition, include the `DISCONNECT FROM SESSION` option so that Redo Apply runs in a background session.

The transmission of redo data to the remote standby location does not occur until after a log switch. Issue the following command on the primary database to force a log switch:

```
SQL> ALTER SYSTEM SWITCH LOGFILE;
```

Special Note: Standby Database on the Same System



- Standby database data files must be at a different location.
- Each database instance must archive to different locations.
- Service names must be unique.
- This standby database does not protect against disaster.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you have a standby database on the same system as the primary database, you must use the following guidelines:

- The data files must be renamed. The actual file names can be the same, but at least the directory path must be different. This means that you must use the `DB_FILE_NAME_CONVERT` and `LOG_FILE_NAME_CONVERT` parameters.
Note: If the standby database uses Oracle Managed Files (OMF), do not set the `DB_FILE_NAME_CONVERT` or `LOG_FILE_NAME_CONVERT` parameters.
- If a standby database is located on the same system as the primary database, the archival directories for the standby database must use a different directory structure than the primary database. Otherwise, the standby database may overwrite the primary database files.
- If you do not explicitly specify unique service names and if the primary and standby databases are located on the same system, the same default global name (consisting of the database name and domain name from the `DB_NAME` and `DB_DOMAIN` parameters) will be in effect for both the databases.
- If the standby database is on the same system as the primary database, it does not protect against disaster. A disaster is defined as total loss of the primary database system. If the standby database is on the same system, it will be lost as well. *This configuration should be used only for testing and training purposes.*

Preventing Primary Database Data Corruption from Affecting the Standby Database

- Oracle Database processes can validate redo data before it is applied to the standby database.
- Corruption detection checks occur on the primary database during redo transport and on the standby database during redo apply.
- Implement lost write detection by setting `DB_LOST_WRITE_PROTECT` to `TYPICAL` on the primary and standby databases.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Data Guard uses Oracle processes to validate redo data before it is applied to the standby database.

Corruption-detection checks occur at the following key interfaces:

- On the primary database during redo transport by the LGWR, LNS, and ARCn processes
- On the standby database during redo apply by the RFS, ARCn, MRP, and DBWn processes

If redo corruption is detected by Redo Apply at the standby database, Data Guard will re-fetch valid logs as part of archive log gap handling.

A lost write occurs when an I/O subsystem acknowledges the completion of a write but the write did not occur in persistent storage. On a subsequent block read, the I/O subsystem returns the stale version of the data block, which is used to update other blocks of the database, thereby corrupting the database.

Set the `DB_LOST_WRITE_PROTECT` initialization parameter on the primary and standby databases to enable the database server to record buffer cache block reads in the redo log so that lost writes can be detected.

You can set DB_LOST_WRITE_PROTECT as follows:

- **TYPICAL on the primary database:** The instance logs buffer cache reads for read/write tablespaces in the redo log
- **FULL on the primary database:** The instance logs reads for read-only tablespaces as well as read/write tablespaces
- **TYPICAL or FULL on the standby database or on the primary database during media recovery:** The instance performs lost write detection
- **NONE on either the primary database or the standby database (the default):** No lost write detection functionality is enabled

When a standby database applies redo during managed recovery, it reads the corresponding blocks and compares the system change numbers (SCNs) with the SCNs in the redo log before doing the following:

- If the block SCN on the primary database is lower than on the standby database, it detects a lost write on the primary database and returns an external error (ORA-752).
- If the SCN is higher, it detects a lost write on the standby database and returns an internal error (ORA-600 3020).

In both cases, the standby database writes the reason for the failure in the alert log and trace file.

The recommended procedure to repair a lost write on a primary database is to fail over to the physical standby and re-create the primary. To repair a lost write on a standby database, you must re-create the standby database or affected files.

Summary

In this lesson, you should have learned how to:

- Enable FORCE LOGGING
- Create standby redo logs
- Set initialization parameters on the primary database to support the creation of the physical standby database and role transition
- Configure Oracle Net Services
- Create a physical standby database by using the DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE RMAN command
- Start the transport and application of redo



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

12

Oracle Data Guard Broker: Overview

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to describe:

- The Data Guard broker architecture
- Data Guard broker components
- Benefits of the Data Guard broker
- Data Guard broker configurations
- How to use Enterprise Manager to manage your Data Guard configuration
- How to invoke DGMGRL (the Data Guard command-line interface) to manage your Data Guard configuration



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Data Guard Broker: Features

- The Oracle Data Guard broker is a distributed management framework.
- The broker automates and centralizes the creation, maintenance, and monitoring of Data Guard configurations.
- With the broker, you can perform all management operations locally or remotely with easy-to-use interfaces:
 - Oracle Enterprise Manager Grid Control
 - DGMGRL (a command-line interface)



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following are some of the operations that the broker automates and simplifies:

- Automated creation of Data Guard configurations incorporating a primary database, a new or existing standby database, redo transport services, and log apply services
Note: Any of the databases in the configuration can be a Real Application Clusters (RAC) database.
- Adding up to eight new or existing standby databases to each existing Data Guard configuration, for a total of one primary database and from one to nine standby databases in the same configuration
- Managing an entire Data Guard configuration (including all databases, redo transport services, and log apply services) through a client connection to any database in the configuration
- Invoking switchover or failover with a single command to initiate and control complex role changes across all databases in the configuration
- Monitoring the status of the entire configuration, capturing diagnostic information, reporting statistics (such as the log apply rate and the redo generation rate), and detecting problems quickly with centralized monitoring, testing, and performance tools

Data Guard Broker: Components

- Client-side:
 - Oracle Enterprise Manager Grid Control
 - DGMGRL (command-line interface)
- Server-side: Data Guard monitor
 - DMON process
 - Configuration files



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Oracle Data Guard broker consists of both client-side and server-side components.

On the client, you can use the following Data Guard components to define and manage a configuration:

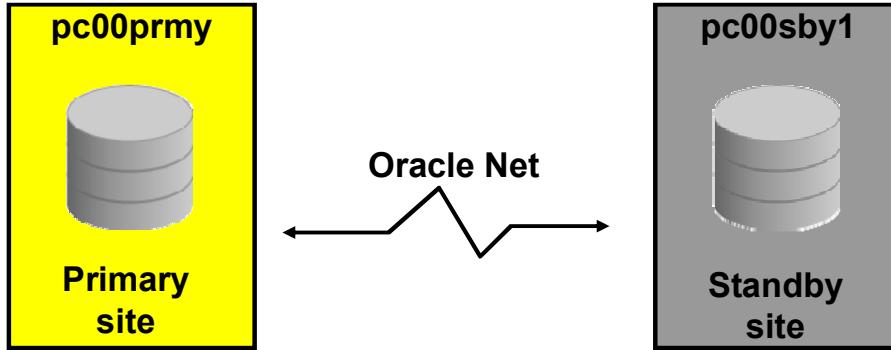
- Oracle Enterprise Manager
- DGMGRL, which is the Data Guard command-line interface (CLI)

On the server, the Data Guard monitor is a broker component that is integrated with the Oracle database. The Data Guard monitor comprises the Data Guard monitor (DMON) process and broker configuration files, with which you can control the databases of that configuration, modify their behavior at run time, monitor the overall health of the configuration, and provide notification of other operational characteristics.

The configuration file contains profiles that describe the current state and properties of each database in the configuration. Associated with each database are various properties that the DMON process uses to control the database's behavior. The properties are recorded in the configuration file as a part of the database's object profile that is stored there. Many database properties are used to control database initialization parameters related to the Data Guard environment.

Data Guard Broker: Configurations

The most common configuration is a primary database at one location and a standby database at another location.



ORACLE

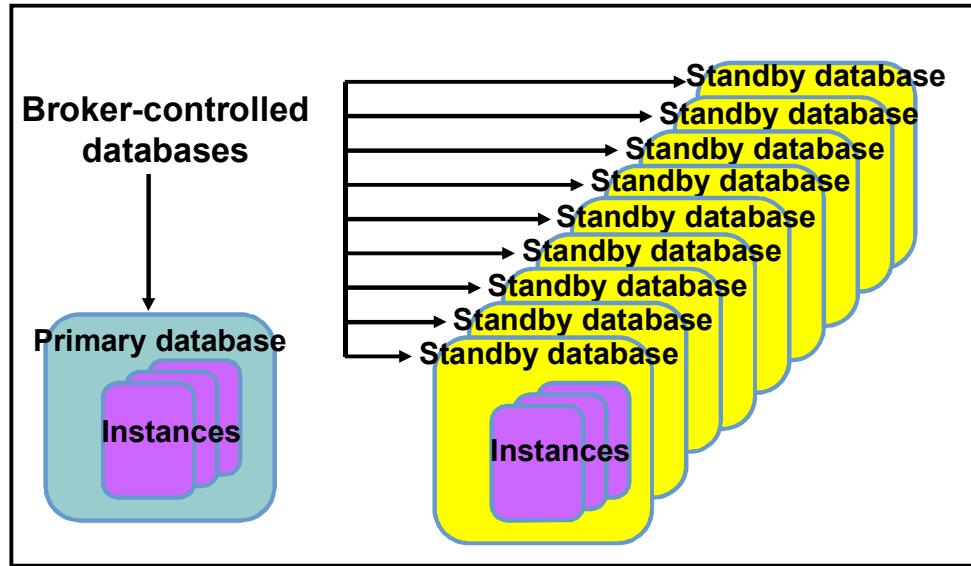
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A Data Guard configuration consists of one primary database and up to nine standby databases. The databases in a Data Guard configuration are typically dispersed geographically and are connected by Oracle Net.

A Data Guard broker configuration is a logical grouping of the primary and standby databases in a Data Guard configuration. The broker's DMON process configures and maintains the broker configuration components as a unified group of resource objects that you can manage and monitor as a single unit.

Data Guard Broker: Management Model

Data Guard Broker Configuration



ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Data Guard broker performs operations on the following logical objects:

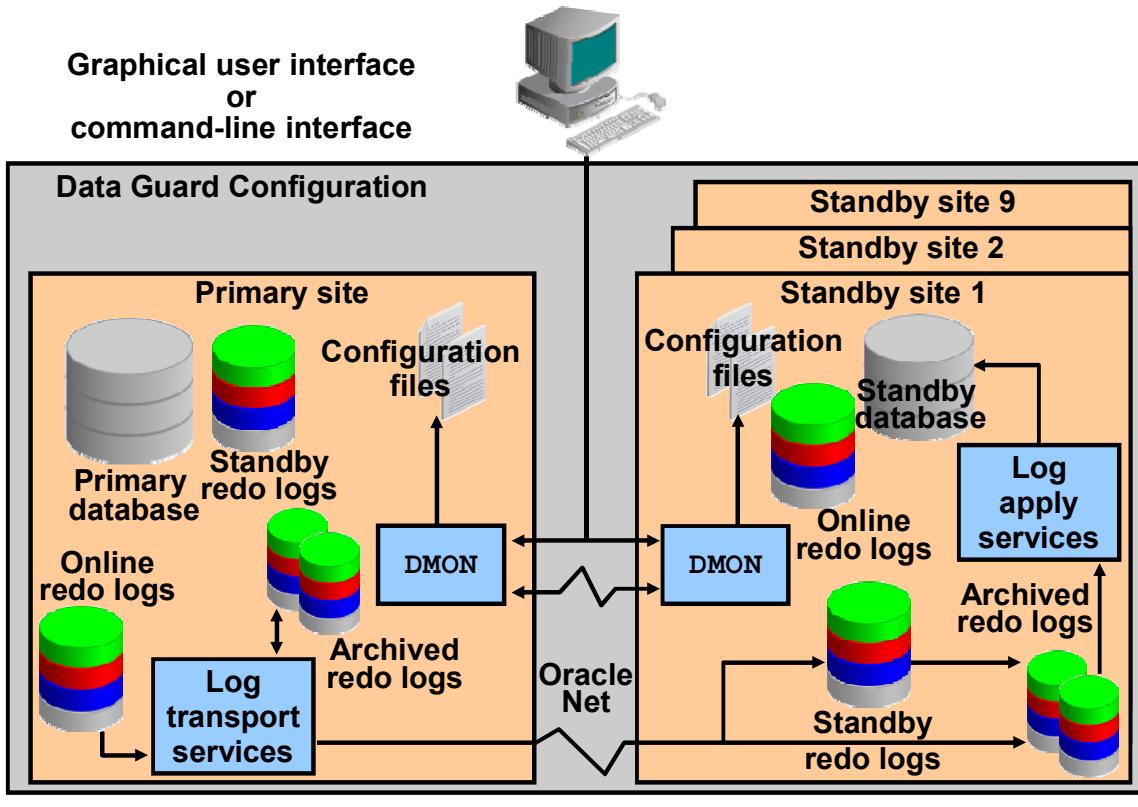
- Configuration of databases
- Single database

A broker configuration consists of:

- **Configuration object**
A named collection of database profiles. A database profile is a description of a database object, including its current state, current status, and properties.
- **Database objects**
Objects corresponding to primary or standby databases
- **Instance objects**
A database object may comprise one or more instance objects if it is a RAC database.

The broker supports one or more Data Guard configurations, each of which includes a profile for one primary database as well as profiles for up to nine physical, logical, RAC or non-RAC standby databases.

Data Guard Broker: Architecture



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Data Guard broker helps you create, control, and monitor a Data Guard configuration. This configuration consists of a primary database that is protected by one or more standby databases. After the broker has created the Data Guard configuration, the broker monitors the activity, health, and availability of all systems in that configuration.

The Data Guard monitor process (DMON) is an Oracle background process that runs on every instance that is managed by the broker. When you start the Data Guard broker, a DMON process is created.

When you use Enterprise Manager or the Data Guard command-line interface (CLI), the DMON process is the server-side component that interacts with the local instance and the DMON processes that are running on other sites to perform the requested function. The DMON process is also responsible for monitoring the health of the broker configuration and for ensuring that every instance has a consistent copy of the configuration files in which the DMON process stores its configuration data. There are two multiplexed versions of the configuration file on each instance.

Data Guard Monitor: DMON Process

- Server-side background process
- Part of each database instance in the configuration
- Created when you start the broker
- Performs requested functions and monitors the resource
- Communicates with other DMON processes in the configuration
- Updates the configuration file
- Creates the `drc<SID>` trace file in the location set by the `DIAGNOSTIC_DEST` initialization parameter
- Modifies initialization parameters during role transitions as necessary



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Data Guard monitor comprises two components: the DMON process and the configuration file.

The DMON process is an Oracle background process that is part of each database instance managed by the broker. When you start the Data Guard broker, a portion of the SGA is allocated and a DMON process is created. The amount of memory allocated is typically less than 50 KB per site; the actual amount on your system varies.

When you use Enterprise Manager or the CLI, the DMON process is the server-side component that interacts with the local instance and the DMON processes running on other sites to perform the requested function.

The DMON process is also responsible for monitoring the health of the broker configuration and for ensuring that every database has a consistent copy of the broker configuration files in which the DMON process stores its configuration data.

Benefits of Using the Data Guard Broker

- Enhances the high-availability, data protection, and disaster protection capabilities inherent in Oracle Data Guard by automating both configuration and monitoring tasks
- Streamlines the process for any one of the standby databases to replace the primary database and take over production processing
- Enables easy configuration of additional standby databases
- Provides simplified, centralized, and extended management
- Automatically communicates between the databases in a Data Guard configuration by using Oracle Net Services
- Provides built-in validation that monitors the health of all databases in the configuration



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

By automating the tasks required to configure and monitor a Data Guard configuration, the broker enhances the high-availability, data protection, and disaster protection capabilities that are inherent in Oracle Data Guard. If the primary database fails, the broker streamlines the process for any one of the standby databases to replace the primary database and take over production processing.

The broker enables easy configuration of additional standby databases. After creating a Data Guard configuration consisting of a primary and a standby database, you can add up to eight standby databases to each Data Guard configuration.

Comparing Configuration Management with and Without the Data Guard Broker

	With the Broker	Without the Broker
General	Manage databases as one	Manage databases separately
Creation of the standby database	Use Grid Control wizards	Manually create files
Configuration and management	Configure and manage from single interface	Set up services manually for each database
Monitoring	<ul style="list-style-type: none"> • Monitor continuously • Unified status and reports • Integrate with EM events 	Monitor each database individually through views
Control	Invoke role transitions with a single command	Coordinate sequences of multiple commands across database sites for role transitions

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide provides an overview of configuration management with and without the Data Guard broker (source: Table 1-1, “Configuration Management With and Without the Broker,” in *Oracle Data Guard Broker*).

Data Guard Broker Interfaces

- Command-line interface (CLI):
 - Is started by entering DGMGRL at the command prompt where the Oracle server or an Oracle client is installed
 - Enables you to control and monitor a Data Guard configuration from the prompt or in scripts
- Oracle Enterprise Manager Grid Control:
 - Provides wizards to simplify creating and managing standby databases



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The DGMGRL command-line interface includes:

- Configuration and setup tasks
- Management and control of the configuration
- Commands to check the status and health of the configuration
- Commands to execute role changes

Oracle Enterprise Manager Grid Control includes the following Data Guard features:

- Wizard-driven creation of standby databases
- Wizard-driven creation of a broker configuration based on an existing primary and standby database
- Complete monitoring and proactive event reporting through email or pagers
- Simplified control of the databases through their potential states. For example, with Enterprise Manager you can start or stop the redo transport services, start or stop the log apply services, and place a standby database in read-only mode.
- “Pushbutton” switchover and failover. Grid Control enables you to execute a switchover or failover between a primary and a standby database by simply clicking a button.

Note: After defining a Data Guard broker configuration, you should use DGMGRL or Enterprise Manager Grid Control to manage your configuration. You should not use SQL commands to manage the databases because you could cause a conflict with the broker.

Using the Command-Line Interface of the Data Guard Broker

```
DGMGRL> connect sys/oracle
Connected.
DGMGRL> show configuration verbose

Configuration
  Name:          DGConfig1
  Enabled:       YES
  Protection Mode: MaxAvailability
  Databases:
    pc00prmy - Primary database
    pc00sby1 - Physical standby database

  Fast-Start Failover: DISABLED

  Current status for "DGConfig1":
  SUCCESS
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

DGMGRL Commands

The following commands are available in DGMGRL (the Data Guard CLI).

Note: Many of these commands have additional arguments that are not described here. See *Oracle Data Guard Broker* for detailed information.

- **ADD:** Adds a standby database to the broker configuration
- **CONNECT:** Connects a given username to the specified instance
- **CREATE:** Enables you to create broker configurations
- **DISABLE:** Enables you to disable broker control of a configuration or database so that the object is no longer managed by the broker
- **EDIT:** Used to edit a configuration, database, or instance
- **ENABLE:** Enables you to enable broker control of a configuration or database
- **EXIT/QUIT:** Exits DGMGRL
- **FAILOVER:** Performs a database failover operation in which one of the standby databases changes to the role of primary database (This is an unplanned transition that may result in the loss of application data.)
- **HELP:** Displays online help for the commands in DGMGRL
- **REINSTATE:** Changes a disabled database into a viable standby database
- **REMOVE:** Removes a broker configuration, including all of its database profiles, a specified standby database profile, or knowledge of an instance

- **SHOW:** Displays either a brief or a detailed summary of information about the broker configuration, database, or instance; can also display the dependency tree and default online states for the broker configuration, as well as the configuration log or the Oracle database alert log
- **SHUTDOWN:** Shuts down a currently running Oracle database instance
- **START:** Starts the Fast-Start Failover Observer
- **STARTUP:** Starts an Oracle instance with several options, including mounting and opening a database
- **STOP:** Stops the Fast-Start Failover Observer
- **SWITCHOVER:** Performs a switchover operation in which the current primary database becomes a standby database and the standby database to which the CLI is currently connected becomes the primary database

Using Oracle Enterprise Manager 10g Grid Control

The Administration tab displays links that allow you to administer database objects and initiate displays links that provide functions that control the flow of data between or outside Oracle data

High Availability

Backup/Recovery	Backup/Recovery Settings
Schedule Backup	Backup Settings
Perform Recovery	Recovery Settings
Manage Current Backups	Recovery Catalog Settings
Manage Restore Points	
Backup Reports	

Data Guard

Setup and Manage	← Click “Setup and Manage” to access the Data Guard pages.
----------------------------------	--

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To access the Data Guard features in Grid Control:

1. Click the Targets tab to go to the Targets page.
2. Click Databases to go to the Databases page, where you can see a list of all discovered databases, including the primary database.
3. Click the primary database to go to the primary database home page.
4. Click Maintenance.
5. Click “Setup and Manage” in the Data Guard section of the Maintenance page to open the Data Guard Overview page.

Data Guard Overview Page

Database Instance: pc00prmy.us.oracle.com >

Data Guard

Page Refreshed February 15, 2008 1:48:57 PM EST View Data

Overview

Data Guard Status	✓ Normal
Protection Mode	Maximum Availability
Fast-Start Failover	Disabled

Primary Database

Name	pc00prmy.us.oracle.com
Host	edt3r17p0.us.oracle.com
Data Guard Status	✓ Normal
Current Log	38
Properties	Edit

Standby Progress Summary

The transport lag is the time difference between the primary and the standby. The apply lag is the time difference between the primary and the standby.

seconds

1.0
0.5
0.0

0 0

pc00sby1.us.oracle.com

Standby Databases

		Edit	Remove	Switchover	Failover	
Select	Name	Host	Data Guard Status	Role	Last Received Log	Last Applied Log
<input checked="" type="radio"/>	pc00sby1.us.oracle.com	edt3r17p2.us.oracle.com	✓ Normal	Physical Standby	37	36

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On the Data Guard Overview page, you can:

- View the protection mode and access the page to edit the protection mode
- View a summary showing the amount of data that the standby database has not received
- View information about the primary database
- View or access pages to change information for the standby databases:
 - Add a standby database to the broker configuration
 - Change the state or properties
 - Discontinue Data Guard broker control
 - Switch the role from standby to primary
 - Transition the standby database to the role of primary database
- Access pages to view performance information for the configuration and status of online redo log files for each standby database
- Perform a verification process on the Data Guard configuration

Click Help to access information about each page.

Note: You access the Data Guard Overview page by clicking “Setup and Manage” in the Data Guard section of the database Maintenance page.

Benefits of Using Enterprise Manager

- Enables you to manage your configuration by using a familiar interface and event-management system
- Automates and simplifies the complex operations of creating and managing standby databases through the use of wizards
- Performs all Oracle Net Services configuration changes that are necessary to support redo transport services and log apply services
- Provides a verify operation to ensure that redo transport services and log apply services are configured and functioning properly
- Enables you to select a new primary database from a set of viable standby databases



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Managing your Data Guard configuration with Oracle Enterprise Manager Grid Control provides the following benefits:

- Enables you to manage your configuration using the familiar Enterprise Manager interface and event-management system
- Provides a wizard that automates the complex tasks involved in creating a broker configuration
- Provides the Add Standby Database Wizard to guide you through the process of adding more databases
- Performs all Oracle Net Services configuration changes that are necessary to support redo transport services and log apply services across the configuration
- Provides a verify operation to ensure that redo transport services and log apply services are configured and functioning properly
- Enables you to select a new primary database from a set of viable standby databases when you need to initiate a role change for a switchover or failover operation

Summary

In this lesson, you should have learned how to:

- Describe the Data Guard broker management model
- Describe the Data Guard broker architecture
- Describe Data Guard broker components
- Access Enterprise Manager
- Invoke DGMGRL



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

13

Configuring Data Protection Modes

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to do the following:

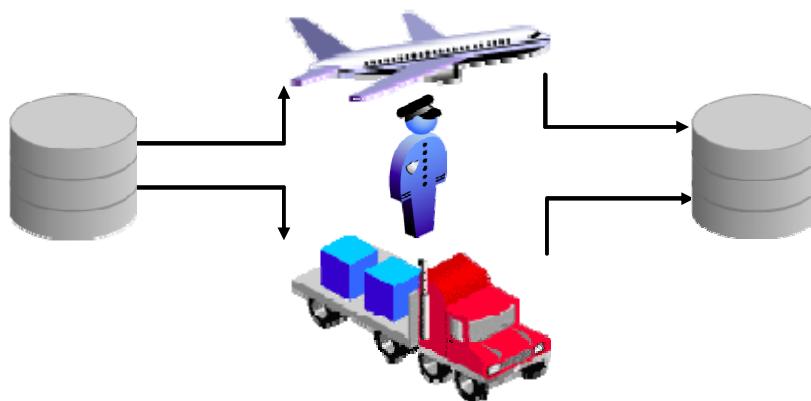
- Describe the data protection modes
- Change the data protection mode of your configuration



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Data Protection Modes and Redo Transport Modes

- A data protection mode requires a specific redo transport mode.
- A redo transport mode alone does not define a data protection mode.



ORACLE

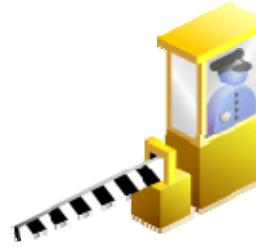
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When you define a redo transport mode, you are configuring the shipment of log files from the primary database to the standby database (physical or logical). You must set your redo transport mode to support the protection mode that you want for your configuration. However, configuring the redo transport mode alone does not set up the protection mode.

After setting up the redo transport mode, you can put the configuration into a data protection mode. The data protection mode setting causes internal rules to be implemented, ensuring that your configuration is protected at the necessary level.

Data Protection Modes

- Three data protection modes:
 - Maximum protection
 - Maximum availability
 - Maximum performance
- Help to balance data availability and system performance



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Data Guard offers maximum protection, maximum availability, and maximum performance modes to help enterprises balance data availability against system performance requirements.

In some situations, a business cannot afford to lose data. In other situations, the availability of the database may be more important than the loss of data. Some applications require maximum database performance and can tolerate the potential loss of data.

Maximum Protection Mode

- Maximum protection mode ensures zero data loss in the event of a failure of the primary database, the network, or all standby databases.
- The primary database shuts down if a fault prevents it from writing its redo stream to at least one synchronized standby database.
- Redo data must be written to both the local online redo log and the standby redo log on at least one synchronized standby database.
- Configuration requirements: At least one standby database must have a standby redo log, and that standby database destination must be configured with the `SYNC` and `AFFIRM` redo transport attributes.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

This protection mode ensures that no data loss occurs if the primary database fails. To provide this level of protection, the redo data that is needed to recover each transaction must be written to both the local online redo log and the standby redo log on at least one standby database before the transaction commits. To ensure that data loss cannot occur, the primary database shuts down if a fault prevents it from writing its redo stream to at least one remote standby redo log. For multiple-instance RAC databases, Data Guard shuts down the primary database if it is unable to write the redo records to at least one properly configured database instance.

Maximum protection mode requirements:

- Configure standby redo log files on at least one standby database.
- Set the `SYNC` and `AFFIRM` attributes of the `LOG_ARCHIVE_DEST_n` parameter for at least one standby database destination.

Note: Oracle recommends a minimum of two standby databases for maximum protection mode.

Maximum Availability Mode

- Maximum availability mode ensures zero data loss without compromising the availability of the primary database.
- Redo data must be written to both the local online redo log and the standby redo log on at least one synchronized standby database.
- The primary database does not shut down if it cannot write to at least one synchronized standby database.
- If no synchronized standby databases are available, the primary database operates in an unsynchronized mode until at least one standby database is synchronized.
- Configuration requirements: At least one standby database must have a standby redo log, and that standby database destination must be configured with the SYNC and AFFIRM redo transport attributes.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

This protection mode provides the highest possible level of data protection without compromising the availability of the primary database. A transaction does not commit until the redo that is needed to recover that transaction is written to the local online redo log and to at least one remote standby redo log. The primary database does not shut down if a fault prevents it from writing its redo stream to a remote standby redo log. Instead, the primary database operates in an unsynchronized mode until the fault is corrected and all gaps in redo log files are resolved. When all gaps are resolved and the standby database is synchronized, the primary database automatically resumes operating in maximum availability mode.

This mode guarantees that no data loss occurs if the primary database fails—but only if a second fault does not prevent a complete set of redo data from being sent from the primary database to at least one standby database.

Maximum availability mode requirements:

- Configure standby redo log files on at least one standby database.
- Set the SYNC and AFFIRM attributes of the LOG_ARCHIVE_DEST_n parameter for at least one standby database.

Maximum Performance Mode

- Maximum performance mode is the default level of data protection.
- This mode provides the highest possible level of data protection without affecting the performance of the primary database.
- Transactions can commit as soon as the redo data is written to the local online redo log.
- Redo data is shipped to the standby database asynchronously with respect to the commitment of the transactions that create the redo data.
- Configuration requirements:
 - Standby redo log on at least one standby database
 - At least one standby database that is configured with the `ASYNC` and `NOAFFIRM` redo transport attributes



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Maximum performance is the default protection mode and provides the highest possible level of data protection without affecting the performance of the primary database. This is accomplished by allowing a transaction to commit as soon as the redo data needed to recover that transaction is written to the local online redo log. The primary database's redo data is also written to at least one standby database, but that redo data is written asynchronously with respect to the commitment of the transactions that create the redo data.

When network links with sufficient bandwidth are used, this mode provides a level of data protection that approaches that of maximum availability mode with minimal impact on primary database performance.

Maximum performance mode requirement: Set the `ASYNC` and `NOAFFIRM` redo transport attributes of the `LOG_ARCHIVE_DEST_n` parameter on at least one standby database.

Comparing Data Protection Modes

Mode	Risk of Data Loss	Transport	If no acknowledgment is received:
Maximum Protection	Zero data loss Double failure protection	SYNC	Stall primary until an acknowledgment is received
Maximum Availability	Zero data loss	SYNC	Stall primary until threshold period expires, then resume processing
Maximum Performance	Potential for minimal data loss	ASYNC	Primary never waits for standby acknowledgment



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Consider the characteristics of each protection mode. You must balance cost, availability, performance, and transaction protection when choosing the protection mode.

Note: If you plan to enable fast-start failover, you must set the protection mode to maximum availability or maximum performance.

Setting the Data Protection Mode by Using DGMGRL

1. Configure standby redo logs.
2. Set the LogXptMode property (if necessary).
 - Maximum protection: SYNC
 - Maximum availability: SYNC
 - Maximum performance: ASYNC
3. Set the data protection mode.

```
DGMGRL> EDIT DATABASE 'pc00sby1' SET PROPERTY  
'LogXptMode'='SYNC';  
Property "LogXptMode" updated  
DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS  
MAXAVAILABILITY;  
Succeeded.
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

1. If you are setting the protection mode to maximum protection or maximum availability, ensure that standby redo log files are configured on the standby database. You must also configure standby redo log files for the primary database or another standby database in the configuration to ensure that it can support the chosen protection mode after a switchover.
2. Use the EDIT DATABASE SET PROPERTY command to set the redo transport mode for the standby database. For example, if you are changing the data protection mode to maximum availability, use the EDIT DATABASE command to specify SYNC for redo transport services:

```
DGMGRL> EDIT DATABASE 'DR_Sales' SET PROPERTY  
'LogXptMode'='SYNC';
```

You must also set the redo transport services for the primary database or another standby database in the configuration to ensure that it can support the chosen protection mode after a switchover.
3. Use the EDIT CONFIGURATION SET PROTECTION MODE AS command to set the overall configuration protection mode. To set the protection mode to maximum availability, use the following command:

```
DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS  
MAXAVAILABILITY;
```

Setting the Data Protection Mode

The screenshot shows the Oracle Enterprise Manager 10g Grid Control interface. The top navigation bar includes links for Hosts, Databases, Application Servers, Web Applications, and Services. Below the navigation is a breadcrumb trail: Database Instance: pc00prmy.us.oracle.com > Data Guard. A message indicates the page was refreshed on November 5, 2007, at 3:58:51 PM EST.

Overview

Data Guard Status	✓ Normal
Protection Mode	Maximum Performance
Fast-Start Failover	Disabled

Primary Database

Name	pc00prmy
Host	edt3r17p0.us.oracle.com
Data Guard Status	✓ Normal
Current Log	57
Properties	Edit

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If the data protection mode that you need requires a standby database to use the SYNC or ASYNC redo transport mode, Enterprise Manager automatically sets the redo transport mode for the primary database and the selected standby databases.

Enterprise Manager automatically determines the correct number and size of standby redo log files that are needed for all databases in the configuration and adds those log files using the directory locations that you specify.

To set the data protection mode with Enterprise Manager:

1. Navigate to the Data Guard page.
2. Click the link in the Protection Mode field to access the Change Protection Mode: Select Mode page.

Setting the Data Protection Mode

The screenshot shows the Oracle Enterprise Manager 10g Grid Control interface. The top navigation bar includes links for Home, Targets, Deployments, Alerts, and Configuration. Below the navigation is a menu bar with Hosts, Databases, Application Servers, Web Applications, Services, Systems, Groups, and All Targets. The main content area displays the path Database Instance: pc00prmy.us.oracle.com > Data Guard. A sub-header 'Change Protection Mode: Select Mode' is present. A descriptive text block explains that Data Guard provides multiple protection modes, noting that higher protection modes reduce data loss but may affect primary database performance. It specifies that changing to Maximum Protection or Maximum Availability requires a SYSDBA connection to the primary database and all standby databases to determine if redo log files are needed. Three radio button options are listed: 'Maximum Protection' (not selected), 'Maximum Availability' (not selected), and 'Maximum Performance' (selected). The 'Maximum Performance' option is described as providing high data protection with the ASYNC log transport mode, and noting it can also be used with the ARCH log transport mode.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

3. Select Maximum Protection, Maximum Availability, or Maximum Performance, and click Continue.
4. If prompted, enter the username and password of a user with SYSDBA privileges. Click Login.
5. Select one or more standby databases to support the protection mode that you selected. (If standby redo log files are needed, verify the names of the log files.) Click OK.
6. On the Confirmation page, click Yes.

Summary

In this lesson, you should have learned how to:

- Describe the data protection modes
- Change the data protection mode of your configuration



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

14

Grid Infrastructure Installation

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Perform preinstallation tasks for Grid Infrastructure
- Install Grid Infrastructure
- Verify the installation
- Configure Automatic Storage Management (ASM) disk groups



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Module 1: Grid Infrastructure Preinstallation Tasks



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Shared Storage Planning for Grid Infrastructure

- There are three ways of storing Grid Infrastructure files:
 - A supported Cluster File System (CFS)
 - A certified Network File System (NFS)
 - Automatic Storage Management (ASM)

Storage Option	Voting/ OCR	Oracle Software
Automatic Storage Manager (ASM)	Yes	No
ASM Cluster File System (ACFS)	No	Yes
Oracle Cluster File System (OCFS2)	Yes	Yes
NFS (certified only)	Yes	Yes
Shared disk slices (block or raw devices)	No	No

- New installations on block or raw devices using DBCA or OUI are not allowed.
- When upgrading an existing RAC database, you can use an existing raw or block device partition and perform a rolling upgrade of the installation.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Oracle Clusterware configuration files (voting and Oracle Cluster Registry [OCR]) can be stored on a supported cluster file system or network file system. With the introduction of Oracle 11g Release 2, the ASM option can now be used to store OCR and voting files. In conjunction with ASM Cluster File System, ASM now provides a complete shared storage solution for Oracle RAC. Other vendor Storage Area Network (SAN) solutions fall under the category of shared block devices in the chart mentioned in the slide.

With the release of Oracle Database 11g Release 2, using DBCA or the Universal Installer to store Oracle Clusterware or Oracle Database files directly on block or raw devices is not supported although the command-line interfaces still accept them.

If you intend to upgrade an existing Oracle RAC database, or an Oracle RAC database with ASM instances, then you can use an existing raw or block device partition and perform a rolling upgrade of your existing installation.

Sizing Shared Storage for Oracle Clusterware

- For new installations, OCR and voting disk files can be placed on:
 - Oracle ASM
 - A supported cluster file system
 - An NFS volume mounted by all cluster nodes
- If normal redundancy is used, at least 2 GB on shared storage volumes is needed for OCR and voting files.
- If installing on ASM, you need to have at least:
 - 2 GB for Oracle Clusterware files in three separate failure groups, with at least three physical disks
 - 1 GB of capacity per disk to ensure that there is sufficient space to create Oracle Clusterware files



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For new installations, OCR and voting disk files can be placed either on Oracle ASM, or on a cluster file system or NFS system. Installing Oracle Clusterware files on raw or block devices is no longer supported, unless an existing system is being upgraded. The only use for raw devices is as Oracle ASM disks.

If you use normal redundancy for Oracle Clusterware files, which includes three Oracle Cluster Registry (OCR) locations and three voting disk locations, then you should have at least 2 GB of file space available on shared storage volumes reserved for Oracle Grid Infrastructure files.

If you plan to install on Oracle ASM, then to ensure high availability of OCR or voting disk files on Oracle ASM, you need to have at least 2 GB for Oracle Clusterware files in three separate failure groups, with at least three physical disks. Each disk must have at least 1 GB of capacity to ensure that there is sufficient space to create Oracle Clusterware files.

Managing Voting Disks in ASM

- Each node must be able to access a majority of voting disks; otherwise, it will be evicted from the cluster.
- Voting disks can be stored on an ASM disk group.
 - They are not regular ASM files.
 - Clusterware knows the location in case ASM is unavailable.
- The number of voting disks is determined by the ASM disk group redundancy setting.
 - 1 for external redundancy disk group
 - 3 for normal redundancy disk group
 - 5 for high redundancy disk group
- A separate failure group is required for each voting disk.
- Voting disks are managed using the `crsctl` utility.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Clusterware uses voting disk files, also called voting disks, to determine which nodes are members of a cluster and to maintain the integrity of the cluster. If you configure voting disks on ASM, then you do not need to manually configure the voting disks. Depending on the redundancy of the specified disk group, a predefined number of voting disks are created.

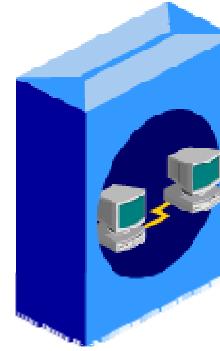
ASM manages voting disks differently from other files that it stores. When you initially configure Oracle Clusterware, you specify the disk group to contain the voting disks. Each voting disk is housed in a separate ASM failure group. You must specify enough failure groups to support the number of voting disks associated with each disk group redundancy setting. For example, you must have at least three failure groups to store your voting disks in a normal redundancy disk group. Voting disks do not appear as regular files within ASM, rather Clusterware records exactly where the voting disk information is located. This arrangement exists so that if ASM is unavailable for any reason, then Cluster Synchronization Services can still access the voting disks and maintain the cluster.

One of the benefits of using an ASM disk group, with either normal or high redundancy, is that if a disk containing a voting disk fails, then as long as there is another disk available in the disk group, ASM automatically recovers the voting disk. Voting disks are managed using the `crsctl` utility. For example, the following command migrates voting disks from their current location to an ASM disk group named `VOTE`:

```
# crsctl replace votedisk +VOTE
```

Oracle Grid Infrastructure 11g Installation

1. Check system requirements.
2. Check network requirements.
3. Install required operating system packages.
4. Set kernel parameters.
5. Create groups and users.
6. Create required directories.
7. Configure installation owner shell limits.
8. Install Oracle Clusterware.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To successfully install Oracle Grid Infrastructure, it is important that you have an understanding of the tasks that must be completed and the order in which they must occur. Before the installation can begin in earnest, each node that is going to be part of your cluster installation must meet the hardware and software requirements that are covered in this lesson. You must perform step-by-step tasks for hardware and software verification, as well as for platform-specific preinstallation procedures. You must install the operating system patches required by the cluster software and verify that the kernel parameters are correct for your needs.

Oracle Grid Infrastructure must be installed by using the graphical OUI. Character-based tool installations are not possible; however, OUI can be run in silent batch mode using response files to supply the values that the installation would need. Ensure that your cluster hardware is functioning normally before you begin this step. Failure to do so results in an aborted or nonoperative installation.

If you intend to use Enterprise Manager Grid Control to manage your cluster deployments, you must next install the Enterprise Manager (EM) agent on each cluster node.

Note: This lesson provides details about performing an installation, but it should not be used as a substitute for the *Installation* manual for your platform.

Checking System Requirements

- At least 2.5 GB of physical memory is needed.

```
# grep MemTotal /proc/meminfo  
MemTotal:        2897320 kB
```

- Set swap space to 2 times the amount of RAM for systems with 4 to 8 GB of RAM and 1.5 times for systems up to 32 GB.

```
# grep SwapTotal /proc/meminfo  
SwapTotal:        4194296 kB
```

- The local /tmp directory should have at least 1 GB free.

```
# df -h /tmp  
Filesystem      Size  Used Avail Use% Mounted on  
/dev/sda6       9.7G  2.3G  7.0G  25%  /
```

- At least 6.5 GB of local storage for the software is needed.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The system must meet the following minimum hardware requirements:

- The minimum required RAM is 2.5 GB for Grid Infrastructure. To determine the amount of physical memory, enter the following command:

```
# grep MemTotal /proc/meminfo
```
- The minimum required swap space is 2 times the amount of RAM for systems with 4 to 8 GB of RAM and 1.5 times total RAM for systems with 8 to 32 GB of RAM. For systems with 4 GB or less of RAM, use swap space equal to RAM. To determine the size of the configured swap space, enter the following command:

```
grep SwapTotal /proc/meminfo
```
- At least 1 GB of disk space must be available in the /tmp directory (TMP and TMPDIR variables can force another location to be used). To determine the amount of disk space available in the /tmp directory, enter the following command:

```
df -h /tmp
```
- At least 2 GB of disk space for the Oracle Clusterware configuration files is needed, assuming the standard redundancy of three OCR files and three voting disks. This must be the shared storage accessible by every node.
- At least 6.5 GB for the Grid Infrastructure software home is required on each node.

Note: The values given here are indented for Linux systems. Please refer to the platform specific Grid Infrastructure installation guides for operating systems other than Linux.

Enabling the Name Service Cache Daemon (nscd)

- To allow Clusterware to better tolerate network failures when using NAS or NFS storage, enable nscd:

```
# /sbin/service nscd start
```

- To check to see if nscd is set to load on system startup, use the chkconfig command.

```
# chkconfig --list nscd
nscd 0:off 1:off 2:off 3:on 4:off 5:off 6:off
```

- It should be “on” for run levels 3 and 5.
- To alter the preceding configuration to ensure nscd is on for both run levels, execute the following command:

```
# # chkconfig --level 35 nscd on
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To allow Oracle Clusterware to better tolerate network failures with NAS devices or NFS mounts, enable the Name Service Cache Daemon (nscd). The nscd provides a caching mechanism for the most common name service requests.

To check to see if nscd is set to load when the system is restarted, enter the command chkconfig --list nscd. For example:

```
# chkconfig --list nscd
```

In the example in the slide, nscd is enabled for run level 3, and disabled for run level 5. The nscd should be enabled for both run level 3 and run level 5.

To change the configuration to ensure that nscd is enabled for both run level 3 and run level 5, enter one of the following command as root:

```
# chkconfig --level 35 nscd on
```

Single-Client Access Name for the Cluster

- During Typical installation, you are prompted to confirm the default Single-Client Access Name (SCAN).
 - SCAN is used to connect to databases within the cluster irrespective of which nodes they are running on.
- By default, the name used as the SCAN is also the name of the cluster.
- The default value for the SCAN is based on the local node name.
- The SCAN must be globally unique in your enterprise.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

During Typical installation, you are prompted to confirm the default Single-Client Access Name (SCAN), which is used to connect to the databases within the cluster irrespective of which nodes they are running on. By default, the name used as the SCAN is also the name of the cluster. The default value for the SCAN is based on the local node name. If you change the SCAN from the default, then the name that you use must be globally unique throughout your enterprise.

In a Typical installation, the SCAN is also the name of the cluster. The SCAN and cluster name must be at least one character long and no more than 15 characters in length, must be alphanumeric, and may contain hyphens.

If you require a SCAN that is longer than 15 characters, then be aware that the cluster name defaults to the first 15 characters of the SCAN.

Checking Network Requirements

- Each node must have at least two NICs.
- Interface names must be the same on all nodes.
- Public NIC must support TCP/IP and Private NIC UDP.
- IP addresses are configured using one of the following options:
 - Oracle Grid Naming Service (GNS) using one static address defined during installation
 - Static addresses that network administrators assign on a network domain name server (DNS) for each node
- Public IP must be registered in the domain name server (DNS) or the /etc/hosts file.

```
# cat /etc/hosts
##### Public Interfaces - net0 #####
xxx.xxx.100.11    host01.example.com    host01
xxx.xxx.100.13    host02.example.com    host02
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following is a list of requirements for network configuration:

- Each node must have at least two network interface cards (NICs): one for the public network and one for the private network.
- The network interface names must be the same across all nodes in the cluster.
- On the public network, each NIC must support TCP/IP. On the private network, each NIC must support User Datagram Protocol (UDP). Windows platforms use TCP only.
- If domain name servers (DNS) are being used, the public IP addresses should be registered.
- Ensure that each node is properly identified using the `hostname` and `ifconfig` utilities.
- If the time stamps among the nodes differ significantly, node evictions and reboots can occur. Network Time Protocol (`ntpd`) can be used to synchronize time stamps between cluster nodes.
 - For NTP, the maximum slew rate possible is limited to .5 ms/s as a consequence of the principles on which the NTP protocol and algorithm design are based.

- As a result, the local clock can take a long time to converge to an acceptable offset, about 2,000s for each second the clock is outside the acceptable range. As a result, an adjustment as much as five minutes (300s) will take almost seven days to complete. During this interval, the local clock will not be consistent with other network clocks and the system cannot be used for distributed applications that require correctly synchronized network time. As a result of this behavior, after the clock has been set, it very rarely strays more than 128ms, even under extreme cases of network path congestion and jitter. Sometimes, in particular when ntpd is first started, the error might exceed 128ms. This may on occasion cause the clock to be stepped backward if the local clock time is more than 128s in the future relative to the server. In some applications, this behavior may be unacceptable. If the -x option is included on the command line, the clock will never be stepped and only slew corrections will be used. In practice, this reduces the false alarm rate where the clock is stepped in error to a vanishingly low incidence.
- If NTP is not configured, Oracle Clusterware installs a cluster time daemon, csstd, in observer mode.

You can configure IP addresses with one of the following options:

- Oracle Grid Naming Service (GNS) using one static address defined during installation, with dynamically allocated addresses for all other IP addresses, obtained from your organization's Dynamic Host Configuration Protocol (DHCP) server, and resolved using a multicast domain name server configured within the cluster
- Static addresses that network administrators manually assign on a network domain name server (DNS) for each node

IP Address Requirements with GNS

- If GNS is used, name resolution requests to the cluster are delegated to the GNS, which is listening on the GNS VIP.
- The GNS VIP address is defined in the DNS domain before installation.
- The DNS must be configured to delegate resolution requests for cluster names to the GNS.
- Before installation, the DNS administrator must establish DNS Lookup to direct the DNS resolution of a subdomain to the cluster.
- A DHCP service on the public network is required that allows the cluster to dynamically allocate the VIP addresses as required by the cluster.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you enable Grid Naming Service (GNS), then name resolution requests to the cluster are delegated to the GNS, which is listening on the GNS virtual IP address. You define this address in the DNS domain before installation. The DNS must be configured to delegate resolution requests for cluster names (any names in the subdomain delegated to the cluster) to the GNS. When a request comes to the domain, GNS processes the requests and responds with the appropriate addresses for the name requested.

To use GNS, before installation, the DNS administrator must establish DNS Lookup to direct the DNS resolution of a subdomain to the cluster. If you enable GNS, then you must have a DHCP service on the public network that allows the cluster to dynamically allocate the virtual IP addresses as required by the cluster.

Note: If you have vendor clusterware installed, then you cannot choose to use GNS, because the vendor clusterware does not support it.

IP Address Requirements for Manual Configuration

- If you do not enable GNS, then the public and virtual IP addresses for each node must be static IP addresses.
- The addresses must be configured before Clusterware installation, but not currently in use.
- Public and virtual IP addresses must be on the same subnet.
- The cluster must have the following addresses configured:
 - A public IP address for each node
 - A virtual IP address for each node
 - A private IP address for each node
 - A Single-Client Access Name (SCAN) for the cluster



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you do not enable GNS, the public and virtual IP addresses for each node must be static IP addresses, configured before installation for each node, but not currently in use. Public and virtual IP addresses must be on the same subnet.

Oracle Clusterware manages private IP addresses in the private subnet on interfaces you identify as private during the installation interview. The cluster must have the following addresses configured:

- A public IP address for each node, with the following characteristics:
 - Static IP address
 - Configured before installation for each node, and resolvable to that node before installation
 - On the same subnet as all other public IP, VIP, and SCAN addresses
- A virtual IP address for each node, with the following characteristics:
 - Static IP address
 - Configured before installation for each node, but not currently in use
 - On the same subnet as all other public IP addresses, VIP addresses, and SCAN addresses

- A Single-Client Access Name (SCAN) for the cluster, with the following characteristics:
 - Three Static IP addresses configured on the domain name server (DNS) before installation so that the three IP addresses are associated with the name provided as the SCAN, and all three addresses are returned in random order by the DNS to the requestor
 - Configured before installation in the DNS to resolve to addresses that are not currently in use
 - Given a name that does not begin with a numeral
 - On the same subnet as all other public IP addresses, VIP addresses, and SCAN addresses
 - Conforms with the RFC 952 standard, which allows alphanumeric characters and hyphens (“-”), but does not allow underscores (“_”)
- A private IP address for each node, with the following characteristics:
 - Static IP address
 - Configured before installation, but on a separate private network, with its own subnet, that is not resolvable except by other cluster member nodes

The SCAN is a name used to provide service access for clients to the cluster. Because the SCAN is associated with the cluster as a whole, rather than to a particular node, the SCAN makes it possible to add nodes to or remove nodes from the cluster without needing to reconfigure clients. It also adds location independence for the databases, so that client configuration does not have to depend on which nodes are running a particular database. Clients can continue to access the cluster in the same way as with previous releases, but Oracle recommends that clients accessing the cluster use the SCAN.

Broadcast and Multicast Requirements

- Broadcast communications (ARP and UDP) must work properly across all the public and private interfaces.
- The broadcast must work across any configured VLANs as used by the public or private interfaces.
- The Oracle mDNS daemon uses multicasting on all interfaces to communicate with other nodes in the cluster.
- With Oracle Grid Infrastructure 11.2.0.2 and later releases, multicasting is required on the private interconnect.
- Multicasting must be enabled for the cluster:
 - Across the broadcast domain as defined for the private interconnect
 - On the IP address subnet ranges 224.0.0.0/24 and 230.0.1.0/24



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Broadcast communications (ARP and UDP) must work properly across all the public and private interfaces configured for use by Oracle Grid Infrastructure release 2 patchset 1 (11.2.0.2) and later releases.

The broadcast must work across any configured VLANs as used by the public or private interfaces.

With Oracle Grid Infrastructure 11 Release 2 (11.2), on each cluster member node, the Oracle mDNS daemon uses multicasting on all interfaces to communicate with other nodes in the cluster.

With Oracle Grid Infrastructure release 2 patchset 1 (11.2.0.2) and later releases, multicasting is required on the private interconnect. For this reason, at a minimum, you must enable multicasting for the cluster:

- Across the broadcast domain as defined for the private interconnect
- On the IP address subnet ranges 224.0.0.0/24 and 230.0.1.0/24

You do not need to enable multicast communications across routers.

Interconnect NIC Guidelines

Optimal interconnect NIC settings can vary depending on the driver used. Consider the following guidelines:

- Configure the interconnect NIC on the fastest PCI bus.
- Ensure that NIC names and slots are identical on all nodes.
- Define flow control: receive=on, transmit=off.
- Define full bit rate supported by NIC.
- Define full duplex autonegotiate.
- Ensure compatible switch settings:
 - If 802.3ad is used on NIC, it must be used and supported on the switch.
 - The Maximum Transmission Unit (MTU) should be the same between NIC and the switch.
- Driver settings can change between software releases.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Failure to correctly configure the network interface cards and switches used for the interconnect results in severe performance degradation and possible node evictions or node fencing. If there is a choice between bus standards such as PCI and PCI express, configure the interconnect NIC on the fastest PCI bus. It is a requirement for the NIC devices and switch to autonegotiate to achieve the highest supported bit rate possible. Flow control should be turned on for receive. Cases have occurred where this setting has been altered between driver software updates and changes. Depending on the mode of link aggregation used, specialized support may be needed at the switch. Synchronization between the switch settings and network interface cards is very important.

For Oracle database Real Application Clusters (RAC), the interconnect will be used to transport database block images. An Oracle database block can be sized up to 32 KB, whereas a typical interconnect communication message averages around 200 bytes.

A misconfigured or faulty interconnect can lead to a variety of problems such as:

- Dropped packets and fragments
- Buffer overflows
- Packet reassembly failures or timeouts
- General Tx/Rx errors

Redundant Interconnect Usage

- Multiple interfaces can be configured for Redundant Interconnect Usage.
 - Clusterware creates from one to four highly available IP (HAIP) addresses.
- Optimal interconnect NIC settings can vary depending on the driver used. Consider the following guidelines:
 - Configure the interconnect NIC on the fastest PCI bus.
 - NIC names and slots must be identical on all nodes.
 - Define flow control: receive=on, transmit=off.
 - Define full bit rate supported by NIC.
 - Define full duplex autonegotiate.
 - Ensure compatible switch settings:
 - If 802.3ad is used on NIC, it must be used on the switch.
 - The MTU should be the same between NIC and the switch.
 - Driver settings can change between software releases.



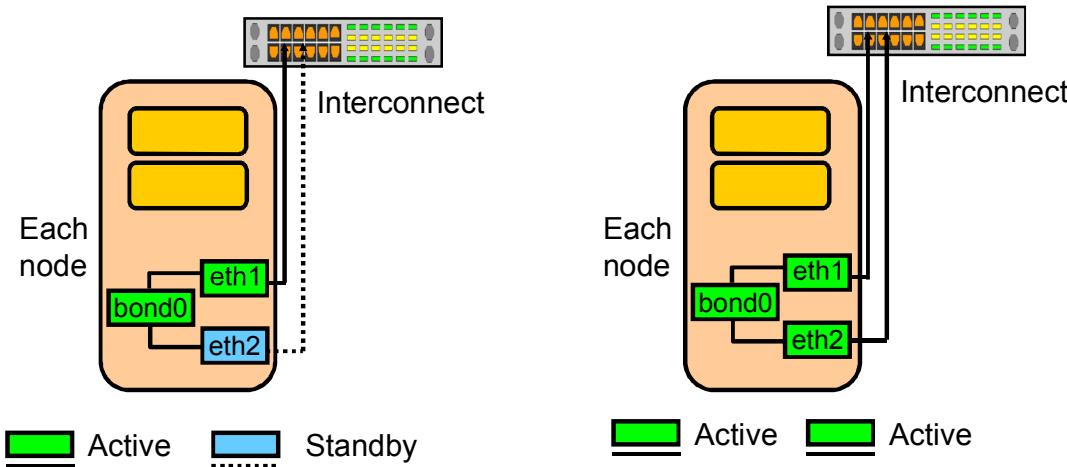
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can define multiple interfaces for Redundant Interconnect Usage by classifying the interfaces as private either during installation or after installation using the `oifcfg setif` command. When you do, Oracle Clusterware creates from one to four (depending on the number of interfaces you define) highly available IP (HAIP) addresses, which Oracle Database and Oracle ASM instances use to ensure highly available and load balanced communications.

The Oracle software [including Oracle RAC, Oracle ASM, and Oracle ACFS, all 11g release 2 (11.2.0.2), or later], by default, uses these HAIP addresses for all of its traffic, allowing for load balancing across the provided set of cluster interconnect interfaces. If one of the defined cluster interconnect interfaces fails or becomes non-communicative, then Oracle Clusterware transparently moves the corresponding HAIP address to one of the remaining functional interfaces.

Interconnect Link Aggregation: Single Switch

- Link aggregation can be used to increase redundancy for higher availability with an Active/Standby configuration.
- Link aggregation can be used to increase bandwidth for performance with an Active/Active configuration.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Interconnect link aggregation involves bonding two or more physical network interface cards into a single logical “bonded” interface. The behavior of the bonded interfaces depends on the settings, modes, and drivers used to accomplish the aggregation.

One strategy often used for highly available configurations is the Active/Standby aggregation, sometimes known as Active/Backup or Active/Passive aggregation. Generally, only one of the network interface cards carries traffic, and the other is available for failover. An example of an Active/Backup setup on Linux as reported with the `ifconfig` command is as follows:

```
bond0 Link encap:Ethernet HWaddr 00:C0:F0:1F:37:B4
      inet addr:XXX.XXX.XXX.YYY Bcast:XXX.XXX.XXX.255
      Mask:255.255.252.0
              UP BROADCAST RUNNING MASTER MULTICAST MTU:1500 Metric:1
              RX packets:7224794 errors:0 dropped:0 overruns:0 frame:0
              TX packets:3286647 errors:1 dropped:0 overruns:1 carrier:0
              collisions:0 txqueuelen:0
```

```
eth1    Link encap:Ethernet  HWaddr 00:C0:F0:1F:37:B4
        inet  addr:XXX.XXX.XXX.YYY  Bcast:XXX.XXX.XXX.255
        Mask:255.255.252.0
              UP BROADCAST RUNNING NOARP SLAVE MULTICAST  MTU:1500  Metric:1
              RX packets:3573025 errors:0 dropped:0 overruns:0 frame:0
              TX packets:1643167 errors:1 dropped:0 overruns:1 carrier:0
              collisions:0 txqueuelen:100
              Interrupt:10 Base address:0x1080

eth2    Link encap:Ethernet  HWaddr 00:C0:F0:1F:37:B4
        inet  addr:XXX.XXX.XXX.YYY  Bcast:XXX.XXX.XXX.255
        Mask:255.255.252.0
              UP BROADCAST RUNNING SLAVE MULTICAST  MTU:1500  Metric:1
              RX packets:3651769 errors:0 dropped:0 overruns:0 frame:0
              TX packets:1643480 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:100
              Interrupt:9 Base address:0x1400
```

The eth1 and eth2 devices are physical network interface cards. The bond0 device is a “virtual” network interface card. Notice that the logical bond0 interface is listed as the MASTER, and the other two interfaces are listed as SLAVE. The interface device without the NOARP (eth2) is the current active SLAVE. Also notice that all three interfaces report the same layer-2 or Media Access Control (MAC) address and have IP addresses. Traffic statistics exist on all Network Interface Card (NIC) devices in this sample output because of extended up time and failures in the past.

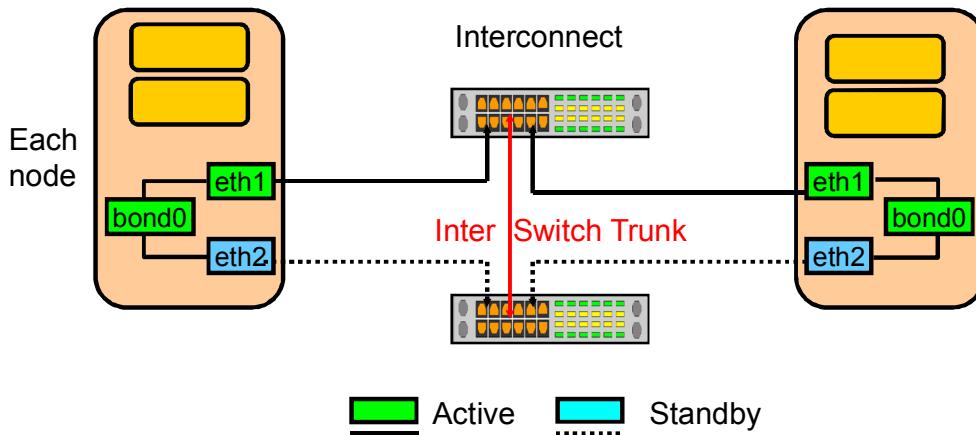
Note: During the installation of the Oracle Clusterware product, the bond0 interface will be supplied as the value to the prompts for the interconnect interface to be used.

Another common strategy for link aggregation involves Active/Active configurations following the IEEE 802.3ad standards. This arrangement involves simultaneous use of both the bonded physical network interface cards in parallel to achieve a higher bandwidth beyond the limit of any one single network card. It is very important that if 802.3ad is used at the NIC layer, the switch must also support and be configured for 802.3ad. Misconfiguration results in poor performance and interface resets or “port flapping.” An alternative is to consider a single network interface card with a higher bandwidth, such as 10 Gb Ethernet instead of 1Gb Ethernet. InfiniBand can also be used for the interconnect.

Link aggregation is sometimes known as “NIC teaming,” “NIC bonding,” “port trunking,” “EtherChannel,” “Multi-Link Trunking (MLT),” “Network Fault Tolerance (NFT),” and “link aggregate group (LAG).” Link aggregation is usually limited to a single switch. Multiswitch solutions include “Split Multi-Link Trunking (SMLT),” “Distributed Split Multi-Link Trunking (DSMLT),” and “Routed Split Multi-Link Trunking (RSMLT).”

Interconnect Link Aggregation: Multiswitch

- Redundant switches connected with an Inter-Switch Trunk may be used for an enhanced highly available design.
- This is the best practice configuration for the interconnect.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

With the single-switch solutions presented in the previous slide, a failure at the switch level would bring down the entire interconnect. A better highly available (HA) design would be to implement a redundant switch strategy as illustrated in the slide, with an Inter-Switch Trunk connecting the switches. This is the best practice design for the Oracle Clusterware interconnect. Only Active/Standby mode is supported in this configuration.

Some of the standard aggregation solutions include:

- Cisco EtherChannel based on 802.3ad
- AIX EtherChannel
- HPUX Auto Port Aggregation
- Sun Trunking, IPMP, GLD
- Linux Bonding (only certain modes)
- Windows NIC teaming

A virtual LAN (VLAN) is supported in a shared switch environment for the interconnect. The interconnect should be a dedicated nonroutable subnet mapped to a single dedicated, nonshared VLAN.

Additional Interconnect Guidelines

UDP socket buffer (rx):

- Default settings are adequate for the majority of customers.
- It may be necessary to increase the allocated buffer size when the:
 - MTU size has been increased
 - netstat command reports errors
 - ifconfig command reports dropped packets or overflow

Jumbo frames:

- Are not an Institute of Electrical and Electronics Engineers (IEEE) standard
- Are useful for Network-Attached Storage (NAS)/iSCSI storage
- Have network device interoperability concerns
- Need to be configured with care and tested rigorously



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The maximum UDP socket receive buffer size varies according to the operating system. The upper limit may be as small as 128 KB or as large as 1 GB. In most cases, the default settings are adequate for the majority of customers. This is one of the first settings to consider if you are receiving lost blocks. Consult the My Oracle Support (formerly, MetaLink) website for best-practice settings for your platform. Three significant conditions that indicate when it may be necessary to change the UDP socket receive buffer size are when the MTU size has been increased, when excessive fragmentation and/or reassembly of packets is observed, and if dropped packets or overflows are observed.

Jumbo frames are not a requirement for Oracle Clusterware and not configured by default. The use of jumbo frames is supported; however, special care must be taken because this is not an IEEE standard and there are significant variances among network devices and switches especially from different manufacturers. The typical frame size for jumbo frames is 9 KB, but again, this can vary. It is necessary that all devices in the communication path be set to the same value.

Note: For Oracle Clusterware, the Maximum Transmission Unit (MTU) needs to be the same on all nodes. If it is not set to the same value, an error message will be sent to the Clusterware alert logs.

Software Requirements (Kernel)

Linux x86 (32-bit and 64-bit) kernel requirements:

Linux Distribution	Requirements
Asianux	Asianux Server 3, Service pack 2
Oracle Enterprise Linux (OEL)	OEL 4 Update 7, kernel 2.6.9 or later OEL 5 Update 2, kernel 2.6.23 or later (32-bit) OEL 5 Update 2, kernel 2.6.19 or later (64-bit) OEL 5 Update 5, with Unbreakable Enterprise Kernel for Linux, kernel 2.6.32-100.0.19 or later (64-bit)
Red Hat Enterprise Linux (RHEL)	RHEL 4 Update 7, kernel 2.6.9 or later RHEL 5 Update 2, kernel 2.6.18 or later RHEL 5, Update 5, with Unbreakable Enterprise Kernel for Linux, 2.6.32 or later (64-bit)
SUSE Enterprise Linux	SUSE 10, kernel 2.6.16.21 or later SUSE 11, kernel 2.6.27.19 or later



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For installations of Oracle Grid Infrastructure, ensure that you have the kernel versions and packages listed in the table in the slide. If you intend to install Oracle Database or Oracle RAC in addition to Oracle Grid Infrastructure, then refer to the *Oracle Grid Infrastructure Installation Guide, 11g Release 2 (11.2) for Linux* to determine whether you must install additional packages for the features you plan to use.

32-Bit Software Requirements: Packages

Linux x86 Oracle 32-bit Grid Infrastructure and Oracle RAC
RPM requirements for Asianux Server 3, OEL 5, and RHEL 5

binutils-2.17.50.0.6	libaio-0.3.106
compat-libstdc++-33-3.2.3	libaio-devel-0.3.106
elfutils-libelf-0.125	libgcc-4.1.2
elfutils-libelf-devel-0.125	libgomp-4.1.2
glibc-2.5-24	libstdc++-4.1.2
glibc-common-2.5	libstdc++-devel-4.1.2
glibc-devel-2.5	make-3.81
glibc-headers-2.5	sysstat-7.0.2
gcc-4.1.2	unixODBC-2.2.11
gcc-c++-4.1.2	unixODBC-devel-2.2.11
kernel-headers-2.6.18	ksh-20060214



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you intend to install an Oracle Database or Oracle cluster database in addition to Oracle Grid Infrastructure, then refer to the list in the slide to determine whether you must install additional packages for the features you plan to use. If you are running Asianux Server 3, OEL5, or RHEL5, refer to the list in the slide for RPM requirements.

To determine whether the required packages are installed, enter a command similar to the following:

```
# rpm -qa | grep package_name
```

It is best to use the grep command to search for the package name without the version information because newer versions may be installed. Install missing packages and their dependencies with the following syntax:

```
# rpm -ivh package_name
```

64-Bit Software Requirements: Packages

Linux x86 Oracle 64-bit Grid Infrastructure and Oracle RAC
RPM requirements for Asianux Server 3, OEL 5, and RHEL 5

compat-libstdc++-33-3.2.3	libaio-0.3.106
compat-libstdc++-33-3.2.3 (32 bit)	libaio-0.3.106 (32 bit)
elfutils-libelf-0.125	libaio-devel-0.3.106
elfutils-libelf-devel-0.125	libgcc-4.1.2
gcc-4.1.2	libgcc-4.1.2 (32 bit)
gcc-c++-4.1.2	libstdc++-4.1.2
glibc-2.5-24	libstdc++-4.1.2 (32 bit)
glibc-2.5-24 (32 bit)	libstdc++-devel 4.1.2
glibc-common-2.5	make-3.81
glibc-devel-2.5	sysstat-7.0.2
glibc-devel-2.5 (32 bit)	unixODBC-2.2.11
glibc-headers-2.5	unixODBC-devel-2.2.11
unixODBC-2.2.11 (32 bit)	ksh-20060214
unixODBC-devel-2.2.11 (32 bit)	binutils-2.17.50.0.6
libaio-devel-0.3.106 (32 bit)	



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For Linuxx86 64-bit systems running Asianux Server 3, OEL 5 Update 2, or RHEL 5 Update 2, the packages listed in the slide (or later versions) must be installed.

Note: Starting with Oracle Grid Infrastructure 11g Release 2 patchset 1 (11.2.0.2), all the 32-bit packages listed in the following table, except for gcc-32bit-4.3, are no longer required for installation. Only the 64-bit packages are required. However, for Oracle 11g release 2 (11.2.0.1), both the 32-bit and 64-bit packages listed in the table are required.

Oracle Validated Configuration RPM

Oracle Validated Configuration RPM for Oracle Linux 5:

- Installs any additional packages needed for installing Oracle Grid Infrastructure and Oracle Database
- Creates an `oracle` user, and the `oralInventory` (`oinstall`) and OSDBA (`dba`) groups
- Sets `limits.conf` and `sysctl.conf` settings, startup parameters, user limits, and driver parameters to values based on recommendations from the Oracle Validated Configurations program



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If your Linux distribution is Oracle Enterprise Linux or Red Hat Enterprise Linux, you can complete most preinstallation configuration tasks by using the Oracle Validated Configuration RPM, available from the Unbreakable Linux Network. When it is installed, the Oracle Validated Configuration RPM does the following:

- Automatically installs any additional packages needed for installing Oracle Grid Infrastructure and Oracle Database
- Creates an `oracle` user and the `oralInventory` (`oinstall`) and OSDBA (`dba`) groups
- Sets and verifies the `sysctl.conf` settings, startup parameters, user limits, and driver parameters to values based on recommendations from the Oracle Validated Configurations program

If you are using Oracle Enterprise Linux 4.7 or later or Oracle Enterprise Linux 5.2, the Oracle Validated Configuration RPM is included on the installation media. Use the following procedure to subscribe to Oracle Unbreakable Linux channels and to add the Oracle Software for Enterprise Linux channel that distributes the Oracle Validated Configuration RPM:

1. Complete a default Oracle Enterprise Linux workstation installation or a default Red Hat Enterprise Linux installation.
2. Register your server with Unbreakable Linux Network. By default, you are registered for the Enterprise Linux Latest channel for your operating system and hardware.

3. Log in to Unbreakable Linux Network at the following URL: <https://linux.oracle.com>.
4. Click the Systems tab, and in the System Profiles list, select a registered server. The System Details window opens and displays the subscriptions for the server.
5. From the Available Channels list, select the `_base` and `_patch` channels corresponding to your Oracle Linux distribution. For example, if your distribution is Oracle Linux 5 Update 5 for x86_64, then select the following:
 - Oracle Linux 5 Update 5 installation media copy (x86_64)
 - Oracle Linux 5 Update 5 Patch (x86_64)
6. Click Subscribe.
7. From a terminal session, as `root`, enter the following command:

```
# up2date --nox --show-channels
```

You should see output indicating that you have subscribed to the Oracle Software for Enterprise Linux channel. For example:

```
e15_u5_i386_base  
e15_u5_x86_64_patch
```

8. Open a terminal session as `root` and install the Oracle Validated Configuration RPM with `up2date`, using the following command:

```
# up2date --install oracle-validated
```

Oracle Linux automatically creates a standard (not role-allocated) Oracle installation owner and groups, and sets up other kernel configuration settings as required for Oracle installations.

9. Repeat steps 1 through 8 on all other servers in your cluster. Check the Oracle Validated Configuration RPM log file to review system configuration changes:

```
/var/log/oracle-validated/results/orakernel.log
```

The Oracle Validated RPM sets kernel parameters and resource limits only for the user account `oracle`. To use multiple software account owners, you must perform system configuration for other accounts manually. In addition, users and groups are created using the next available ID numbers. If server group and user IDs are not identical on the cluster nodes where you run the Oracle Validated RPM, then it is possible that these IDs will be different on the nodes, and the different group and user IDs will cause installation to fail. To avoid this issue, run the command `ID user` on all nodes, where `user` is the software owner user account. If group or user IDs are not identical on all the nodes, then change them as necessary to ensure that they are identical.

Oracle Pre-Install RPM

Oracle RDBMS Pre-Install RPM for Oracle Linux 6:

- Completes most pre-installation configuration tasks
- Is different from Oracle Validated RPM for Oracle Linux 5
- Downloads and installs various software packages and specific versions needed for database installation
- Creates the user `oracle` and the groups `oinstall` and `dba`
- Modifies kernel parameters in `/etc/sysctl.conf`
- Sets hard and soft shell resource limits in `/etc/security/limits.conf`
- Sets `numa=off` in the kernel boot parameters for `x86_64` machines



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Oracle RDBMS Pre-install RPM package (`oracle-rdbms-server-11gR2-preinstall`) is designed specifically for Oracle Linux 6 to aid in the installation of the Oracle Database. You can complete most pre-installation configuration tasks by using this package, which is now available from the Unbreakable Linux Network or from the Oracle Public Yum repository.

This package was formerly known as `oracle-validated`. For Oracle Linux 6 and newer, the name of the package was changed. There will not be an “`oracle-validated`” RPM for Oracle Linux 6, and there will not be an “`oracle-rdbms-preinstall`” RPM for Oracle Linux 5. The key difference between `oracle-validated` and the new pre-install RPM is that the new package sets the bare minimums for Oracle DB 11gR2 installations, instead of the testing maximums set with `oracle-validated`. The new pre-install RPM will configure an Oracle Linux 6 machine so that you can immediately run the OUI database installation.

The pre-install package is available for `x86_64` only. Specifically, the package:

- Causes the download and installation of various software packages and specific versions needed for database installation, with package dependencies resolved via `yum`
- Creates the user `oracle` and the groups `oinstall` and `dba`, which are the defaults used during database installation

Creating Groups and Users

- Create an Oracle Software inventory group on each node.
- Group ID must be consistent on each node.

```
# groupadd -g 501 oinstall
```

- Create the Oracle Software owner on each node.
- User ID must be consistent on each node and the inventory group must be the primary group.
- Most Oracle products (Grid Infrastructure, database, Enterprise Manager, and so on) are usually owned by the same user, typically called `oracle`, but each product can be owned by a different user.

```
# useradd -u 501 -g oinstall oracle  
or  
# useradd -u 502 -g oinstall grid
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

An operating system group needs to be created that will be associated with Oracle Central Inventory (`oraInventory`). `oraInventory` contains a registry of the Oracle home directories from all Oracle products installed on the server. It is designed to be shared among many products. It also contains installation log files and trace files from the installation programs. The suggested name for the operating system group is `oinstall`. In a cluster installation, it is very important that the group ID be the same on all nodes of the cluster.

An operating system user needs to be created to own the Oracle Clusterware installation. Traditionally, all Oracle products installed on the same machine such as clusterware, databases, disk management, and enterprise management tools are owned by the same user called `oracle`. It is possible for each product to be created under a different operating system account. This may be desired if different job responsibilities are used to manage different components. It is very important that the user ID be the same on all nodes of the cluster.

It is required that the Oracle Clusterware software owner and the Oracle Database software owner have the same primary group as the Oracle Inventory group.

Note: If this installation of Oracle Clusterware contains a database and other Oracle products, consider creating the following groups: `dba` and `oper`.

Creating Groups and Users: Example

1

Create Groups:

```
# /usr/sbin/groupadd -g 1000 oinstall
# /usr/sbin/groupadd -g 1100 asmadmin
# /usr/sbin/groupadd -g 1200 dba
# /usr/sbin/groupadd -g 1201 oper
# /usr/sbin/groupadd -g 1300 asmdba
# /usr/sbin/groupadd -g 1301 asmoper
```

2

Create Users

If a single software owner is needed:

```
# /usr/sbin/useradd -u 1100 -g oinstall -G asmdba,dba,
\asmoper oracle
```

If a separate owner for Grid Infrastructure is needed:

```
# /usr/sbin/useradd -u 1100 -g oinstall -G asmdba,dba,
\oracle
# /usr/sbin/useradd -u 1101 -g oinstall -G asmdba,
\asmadmin,asmoper grid
```

3

Create Directories

```
# mkdir -p /u01/app/grid /u01/app/oracle
# chown -R grid:oinstall /u01/app
# chown oracle:oinstall /u01/app/oracle
# chmod 775 /u01/app
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following is an example of how to create the Oracle Inventory group (`oinstall`), OSDBA, OSASM, and OSOPER for ASM groups where they are the same group (`dba`), and how to create the Grid Infrastructure software owner (`grid`) and one Oracle Database owner (`oracle`) with correct group memberships. This example shows how to configure an Oracle base path compliant with the Optimal Flexible Architecture structure with correct permissions:

1. Log in as `root`. Enter commands similar to the example in the slide to create the `oinstall`, `asmadmin`, and `asmdba` groups, and if required, the `asmoper`, `dba`, and `oper` groups. Use the `-g` option to specify the correct group ID for each group.
2. To create the Grid Infrastructure user, enter a command similar to the following (in this example):

```
# /usr/sbin/useradd -u 1100 -g oinstall -G asmdba,dba,asmoper oracle
```

If a separate grid owner is needed:

```
# /usr/sbin/useradd -u 1100 -g oinstall -G asmdba,dba, oracle
# /usr/sbin/useradd -u 1101 -g oinstall -G asmdba,dba,asmoper grid
```

3. Create base directories for Oracle Grid Infrastructure and Database software.

```
# mkdir -p /u01/app/grid /u01/app/oracle
# chown -R grid:oinstall /u01/app
# chown oracle:oinstall /u01/app/oracle
# chmod 775 /u01/app
```

Shell Settings for the Grid Infrastructure User

- Add the following lines to the `/etc/security/limits.conf` file on each node:

```
grid soft nproc 2047
grid hard nproc 16384
grid soft nofile 1024
grid hard nofile 65536
grid soft core unlimited
grid soft core unlimited
oracle soft nproc 2047
oracle hard nproc 16384
oracle soft nofile 1024
oracle hard nofile 65536
oracle soft core unlimited
oracle soft core unlimited
```

- Add to the `/etc/pam.d/login` file on each node:

```
session required pam_limits.so
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Pluggable Authentication Modules (PAM) for Linux is a flexible mechanism for authenticating users and provides a series of library modules that many programs such as `login` and `su` can use for authentication. The advantage is that newer authentication schemes can be plugged in to PAM and made available; you do have to rewrite the programs that depend on authentication. PAM provides for account management, authentication management, password management, and session management. The `/etc/security/limits.conf` file is the configuration file for the `pam_limits` module, one of several PAM modules.

The recommendation for the Oracle Grid Infrastructure owner is to increase the hard limits for the maximum number of processes (`nproc`) and the maximum number of open files (`nofile`) to higher values than the default values. The hard limits are set by the superuser and enforced by the kernel. End-user accounts are not allowed to raise their limits above the hard settings. The soft limits are considered to be the default starting values for normal usage, but end users may alter the soft limits.

Note: These requirements also exist for the account, typically named `oracle`, that will be used for the database software. Consider adding this to `/etc/security/limits.conf` in addition to the Oracle Grid Infrastructure account owner.

Shell Settings for the Grid Infrastructure User

For the Bourne, Bash, or Korn shell, add the following to the /etc/profile file on each node.

```
if [ $USER = "oracle" ] || [ $USER = "grid" ]; then
    if [ $SHELL = "/bin/ksh" ]; then
        ulimit -u 16384
        ulimit -n 65536
    else
        ulimit -u 16384 -n 65536
    fi
    umask 022
fi
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Linux ulimit command provides control over the available system resources to the shell and to processes started by the shell. To check the current settings, issue the following command: ulimit -a.

The maximum number of processes available to a single user is controlled with the -u option, and the maximum number of open file descriptors is set with the -n option. The exact syntax for setting these options depends on the shell that the user account is running under. Oracle recommends that these values be higher than the default values.

Note: These requirements also exist for the account, typically named oracle, that will be used for the database software. Consider adding this to /etc/profile in addition to the Oracle Clusterware account owner.

Module 2: Grid Infrastructure Installation

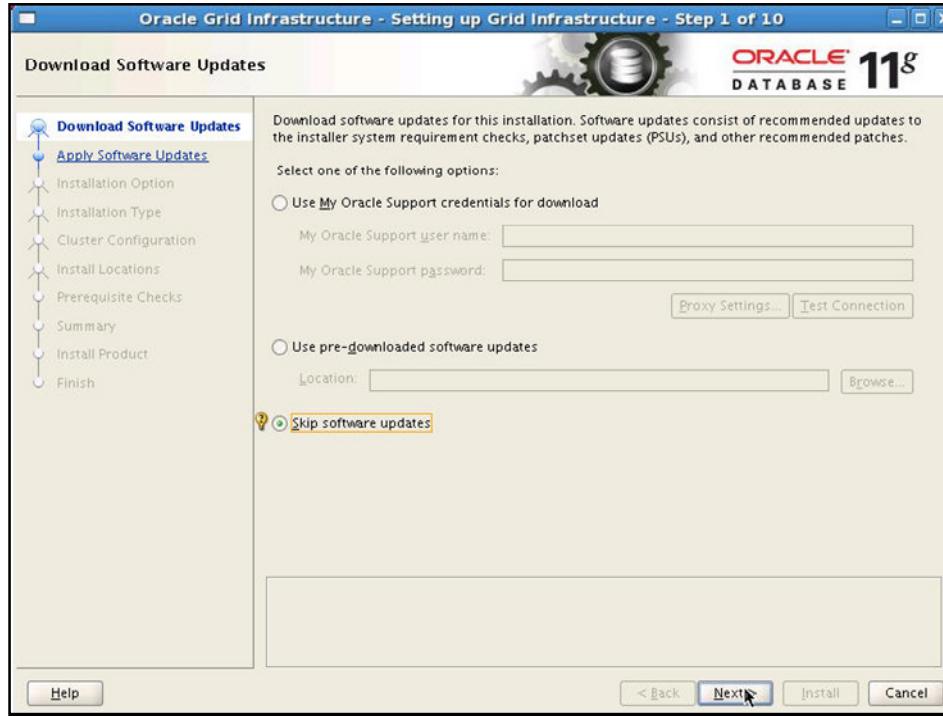
A solid red horizontal bar spanning most of the page width.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Installing Grid Infrastructure

```
./runInstaller
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The OUI has been enhanced to support the new Oracle Real Application Clusters (RAC) improvements found in Oracle Database 11g Release 2. You will immediately notice that the look and feel of the installer is different from past releases.

Run OUI by executing the `runInstaller` command as the grid owner from the installation CD or the staged software directory. If you have any updates or patches to include with the installation, you can specify the appropriate action on the Download Software Updates page. From here, you can provide your Oracle Support login credentials and download patches for this installation. You can also specify a location where updates have been previously staged. In the example in the slide, the user has chosen to skip software updates.

Choosing an Installation Type



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

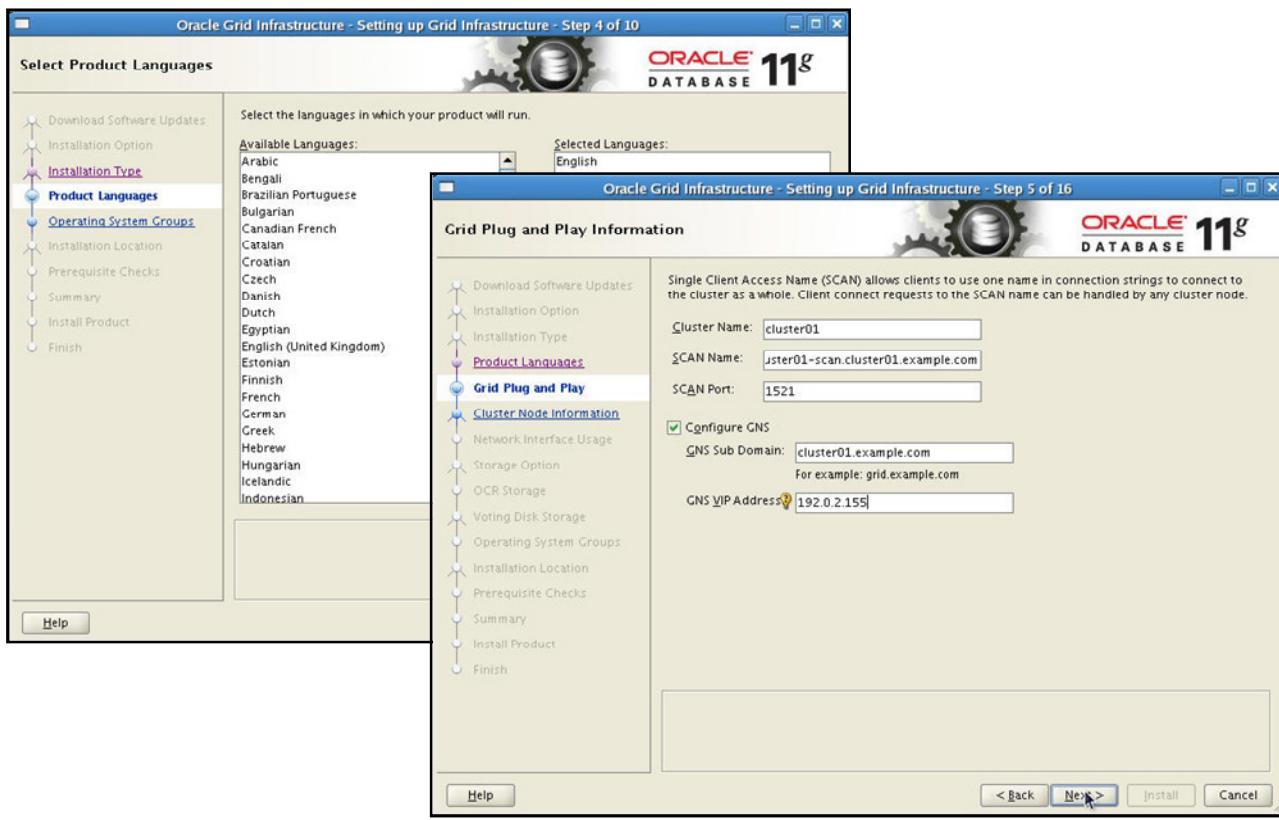
On the Select Installation Option page, click the “Install and Configure Grid Infrastructure For a Cluster” button and click Next

On the Installation Type page, you may choose a Typical Installation employing a basic fixed IP configuration. You may also choose an Advanced Installation incorporating Grid Plug and Play (GPnP) and offering greater flexibility in configuring:

- Shared storage for OCR and voting files
- Network options such as Grid Naming Service (GNS)
- Failure isolation support employing Intelligent Platform Management Interface (IPMI)

Because the Typical Installation option assumes a basic configuration, the interview process is shortened.

Grid Plug and Play Support



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

Grid Plug and Play Support: Advanced Installation

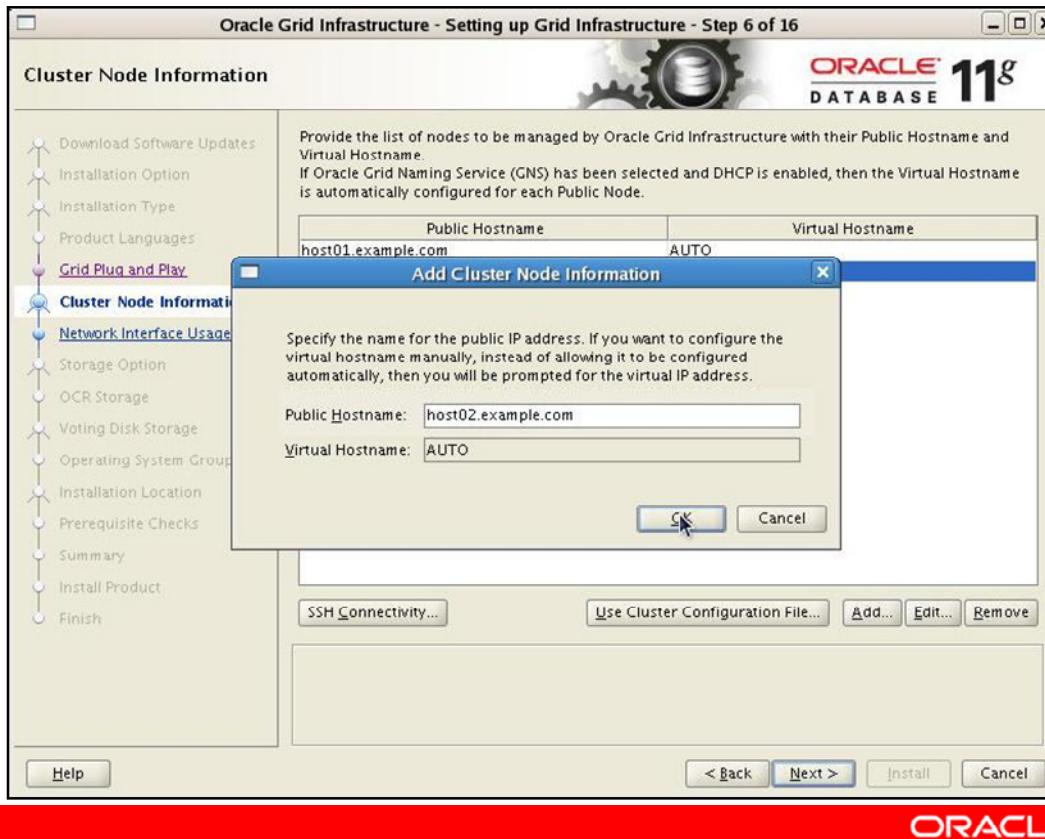
After you have chosen product languages from the Select Product Languages page, the Grid Plug and Play Information page is displayed. You must supply a cluster name. This name should be unique within your subnet. If Grid Control is used, the cluster name should be unique within the Grid Control management realm.

The single-client access name (SCAN) is the address used by clients connecting to the cluster. The SCAN is a domain name registered to three IP addresses, either in the domain name server (DNS) or the Grid Naming Service (GNS). The SCAN addresses need to be on the same subnet as the VIP addresses for the nodes in the cluster. The SCAN domain name must be unique within your corporate network. The SCAN port should default to 1521 unless you have a conflict at that port.

If you specify a GNS subdomain, the SCAN defaults to `clustername-scan.GNSdomain`. For example, if you start the Oracle Grid Infrastructure installation from the `racnode01` server, the cluster name is `cluster01` and the GNS domain is `cluster01.example.com`, then the SCAN is `cluster01-scan.cluster01.example.com`.

If you do not use GNS, the SCAN should be defined in the DNS to resolve to the addresses assigned to that name. Assign three addresses to the SCAN. Finally, you must provide the address of your GNS. This address must be resolvable through your DNS.

Cluster Node Information



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

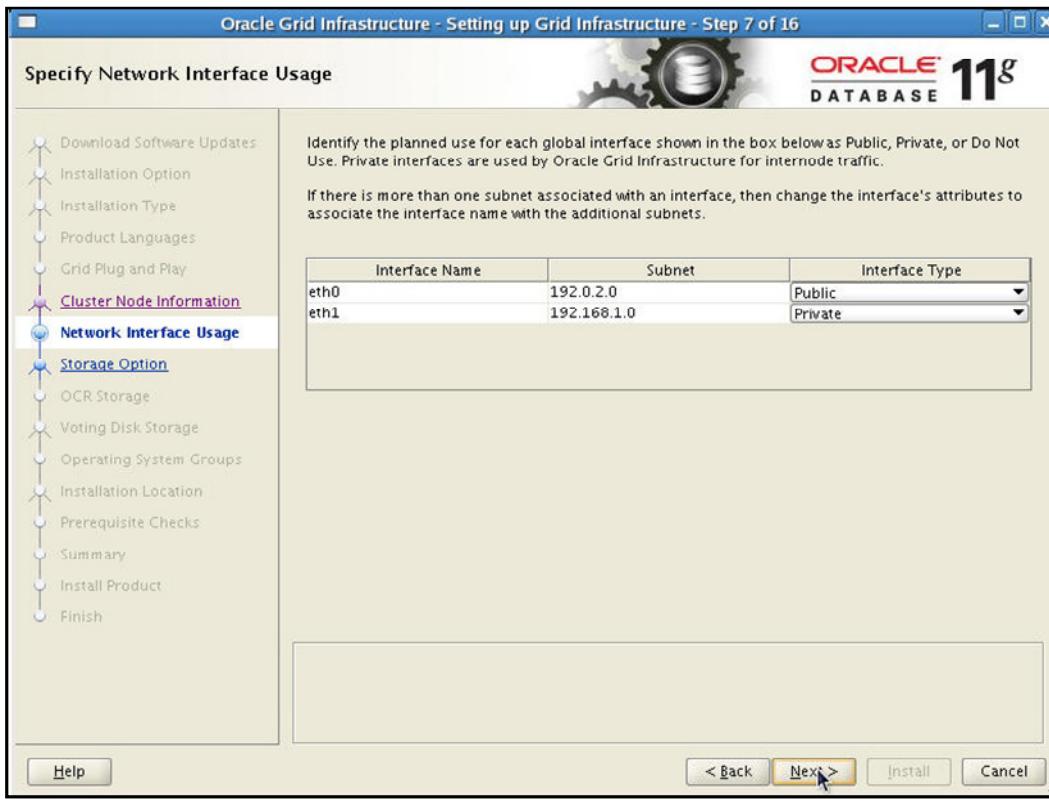
Cluster Node Information: Advanced Installation

On the Cluster Node Information page, the public host names and virtual host names are defined. In the past, the host and VIP names and addresses were defined in the DNS or locally in a hosts file. If you want to configure your cluster in this manner, make sure that the Configure GNS check box is deselected on the previous screen. Click the Add button and enter the Public Node Name and Virtual Host Name for each node in the cluster.

If GNS has been selected on the previous screen and DHCP is configured in the subdomain in which the cluster resides, then configuration is simplified. Click the Add button, and then add the host name as shown in the graphic in the slide. There is no need to provide a Virtual IP name for each node because Virtual IP names are automatically configured by Clusterware using DHCP.

Secure Shell (SSH) can be configured for the specified nodes by clicking the SSH Connectivity button. You will be required to provide the software owners password common to all nodes. When SSH connectivity has been established, click Next to continue.

Specify Network Interface Usage



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

Specify Network Interface Usage: Advanced Installation

On the Specify Network Interface Usage page, you can select the network interfaces on your cluster nodes to use for internode communication. You must choose one interface for the public network and one for the private network. Ensure that the network interfaces that you choose for the interconnect have enough bandwidth to support the cluster and RAC-related network traffic. A gigabit Ethernet interface is highly recommended for the private interconnect. To configure the interface for private use, click the interface type and choose the proper usage for each network interface. In the example shown in the slide, there are two interfaces: eth0 and eth1. The eth0 interface is the hosts' primary network interface and should be marked Public. The eth1 interface is configured for the private interconnect and should be marked Private. If you have other adapters dedicated to a storage network, they should be marked Do Not Use. When you finish, click the Next button to continue.

Storage Option Information

The screenshot shows three panels of the Oracle Database 11g Installation interface:

- Storage Option Information:** Shows two options: "Oracle Automatic Storage Management (Oracle ASM)" (selected) and "Shared File System".
- Create ASM Disk Group:** A dialog box where "Disk Group Name" is set to "DATA". It lists "Candidate Disks" (ORCLASMIDISK01 through ORCLASMIDISK07) and "All Disks".
- OCR Storage Option:** Shows "Normal Redundancy" selected for OCR File Location, with three paths specified: "/OCFS2/vol1/ocr", "/OCFS2/vol2/ocr", and "/OCFS2/vol3/ocr".

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

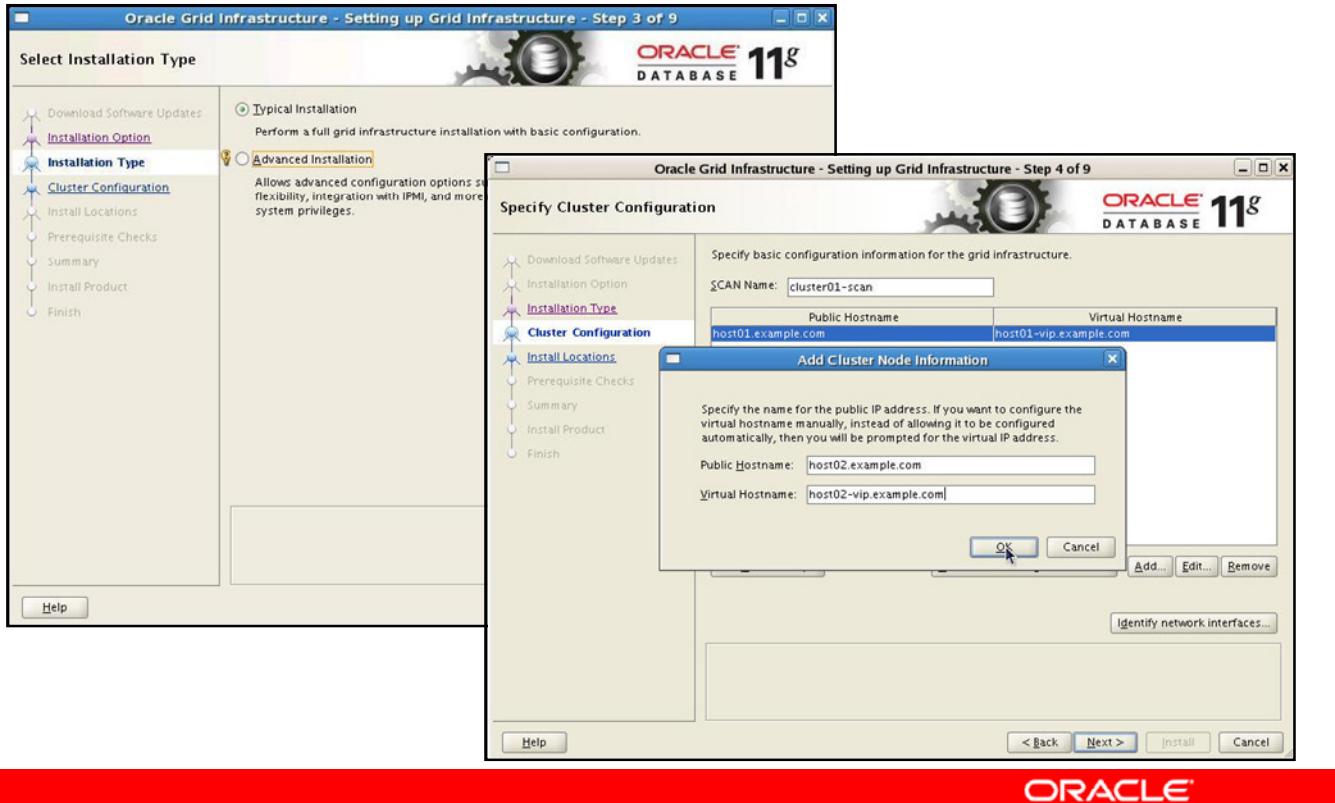
Storage Option Information: Advanced Installation

On the Storage Option Information page, you can select your preferred method for storing Clusterware files. The first option is Automatic Storage Management (ASM). The ability to use ASM disk groups for Clusterware OCR and voting disks is a new feature in Oracle Database 11g Release 2 Grid Infrastructure. If you choose this option and ASM is not yet configured, OUI launches the ASM configuration assistant to configure ASM and a disk group to support the chosen configuration as illustrated in the slide. When using ASM, the redundancy for the OCR and voting files is tied to the redundancy that you define for the disk group and the number of disks you put in the disk group. The voting disk is specifically allocated to three disks in the disk group under normal redundancy and five disks under high redundancy. By default, each disk in the disk group is put in its own failure group and the voting disks are put into different failure groups. The OCR is stored similar to the way database files are stored.

The other storage option available is shared file system storage. Suitable file systems include supported cluster file systems such as OCFS2 or Network File System (NFS). If you select Shared File System on the Storage Option Information page, you will be prompted to provide a cluster file system or NFS directory to be used for the OCR and voting disk files as shown in the slide.

The next page displayed is the Specify ASM Password page. The new ASM instance requires its own SYS user with SYSASM privileges. Specify a secure password here and click Next to continue.

Specify Cluster Configuration: Typical Installation

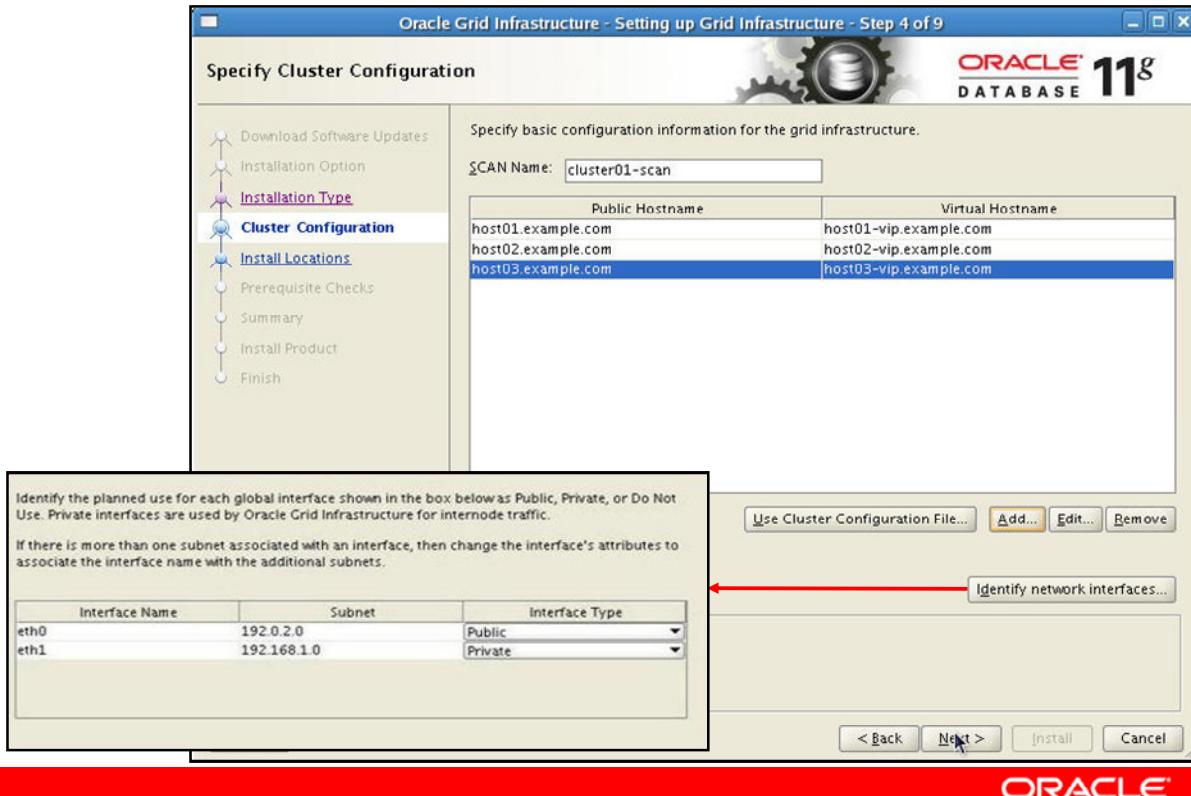


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you choose not to implement GNS, you can choose Typical Installation from the Select Installation Type page. On the Specify Cluster Configuration page, you must enter the SCAN. The SCAN should be unique in your network. In a non-GNS environment, the SCAN should be defined in the DNS or locally in the /etc/hosts file. If the SCAN is defined locally, you are limited to one SCAN. If it is defined in DNS, you can have as many SCANS as you deem necessary, although three SCANS are recommended.

Click the Add button to add cluster node information for the remaining nodes that will be included in this installation. You must provide the host name and VIP name for the additional nodes. Both these names must be resolvable, either locally from the /etc/hosts file or from your DNS.

Specify Cluster Configuration: Typical Installation

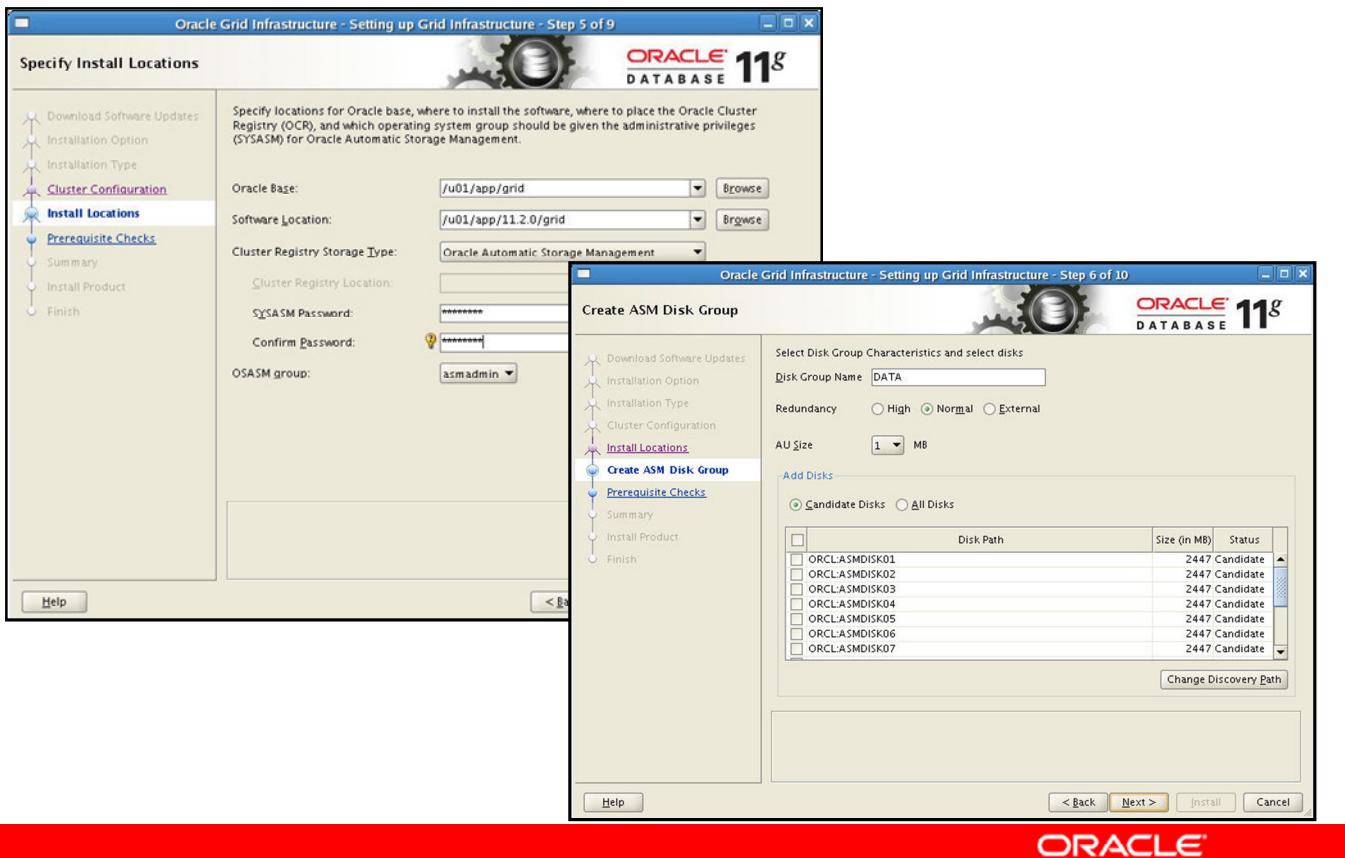


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Secure Shell (SSH) can be configured for the software owner on the specified nodes by clicking the SSH Connectivity button. You will be required to provide the software owners password and it should be identical for all nodes. Click the Setup button to initiate the configuration. If SSH has already been configured for the software owner, you could click the Test button to check the existing configuration.

Click the Specify Network Interfaces button and select the network interfaces on your cluster nodes to use for internode communication. You must choose one interface for the public network and one for the private network. Ensure that the network interfaces that you choose for the interconnect have enough bandwidth to support the cluster and RAC-related network traffic. For each adapter shown, click the interface type and choose the proper usage for each one. In the example shown in the slide, there are two interfaces: eth0 and eth1. The eth0 interface is the hosts' primary network interface and should be marked Public. The eth1 interface is configured for the private interconnect and should be marked Private. When finished, click OK, and then the Next button to continue.

Install Locations: Typical Installation



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

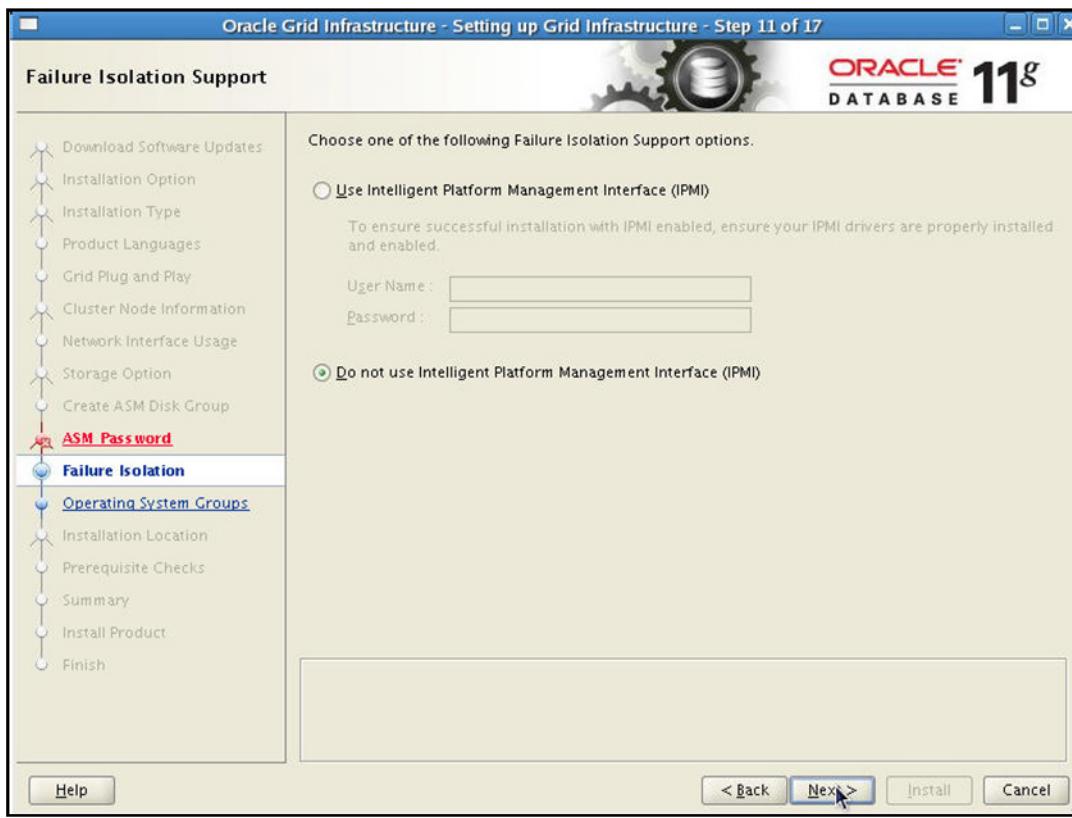
On the Specify Install Locations page, you must enter a base location for installing Oracle software. A separate Oracle base should be created for each operating system user owning an Oracle software installation. In the example in the slide, that location is /u01/app/grid.

In the Software Location field, enter a directory in which to install Oracle Grid Infrastructure. In the example, the grid operating system user will own this installation, so the software location should be /u01/app/11.2.0/grid.

Next, you must indicate where the cluster registry and voting disks will be stored. In the example in the slide, Oracle Automatic Storage Management (ASM) is chosen. If you choose ASM, you must also assign the SYSASM password and the OSASM group. If you choose a cluster file system instead of ASM for the Cluster Registry Storage type, you must provide a valid path to that shared storage. When you have finished, click Next to continue.

If you selected ASM for the Cluster Registry Storage type, you will be prompted to configure an ASM disk group. Click the Change Discovery Path button and provide a path to all disks intended for ASM storage if necessary. Choose a name and desired Redundancy for the disk group. Select the disks to be used for the disk group and click Next to continue.

Failure Isolation Support with IPMI



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

Failure Isolation Support with IPMI: Advanced Installation

Intelligent Platform Management Interface (IPMI) provides a set of common interfaces to computer hardware and firmware that administrators can use to monitor system health and manage the system. With Oracle Database 11g Release 2, Oracle Clusterware can integrate IPMI to provide failure isolation support and to ensure cluster integrity.

You must have the following hardware and software configured to enable cluster nodes to be managed with IPMI:

- Each cluster member node requires a Baseboard Management Controller (BMC) running firmware compatible with IPMI version 1.5, which supports IPMI over local area networks (LANs) and is configured for remote control.
- Each cluster member node requires an IPMI driver installed on each node.
- The cluster requires a management network for IPMI. This can be a shared network, but Oracle recommends that you configure a dedicated network.
- Each cluster node's Ethernet port used by BMC must be connected to the IPMI management network.

If you intend to use IPMI, you must provide an administration account username and password to provide when prompted during installation.

Note: For Oracle Clusterware to communicate with BMC, the IPMI driver must be installed permanently on each node, so that it is available on system restarts.

The IPMI driver is available on the Asianux Linux, Oracle Enterprise Linux, Red Hat Enterprise Linux, and SUSE Enterprise Linux distributions supported with this release.

Privileged Operating System Groups



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Privileged Operating System Groups: Advanced Installation

On the Privileged Operating System Groups page, you must specify the groups that the Grid Infrastructure owner should belong to for proper operating system authentication to ASM. The example in the slide specifies `asmdba` for the ASM Database Administrator (OSDBA) group, `asmoper` for the ASM Instance Administration Operator (OSOPER) group, and `asmadmin` for the ASM Instance Administrator (OSASM) group. Click Next to continue.

Installation and Inventory Locations



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

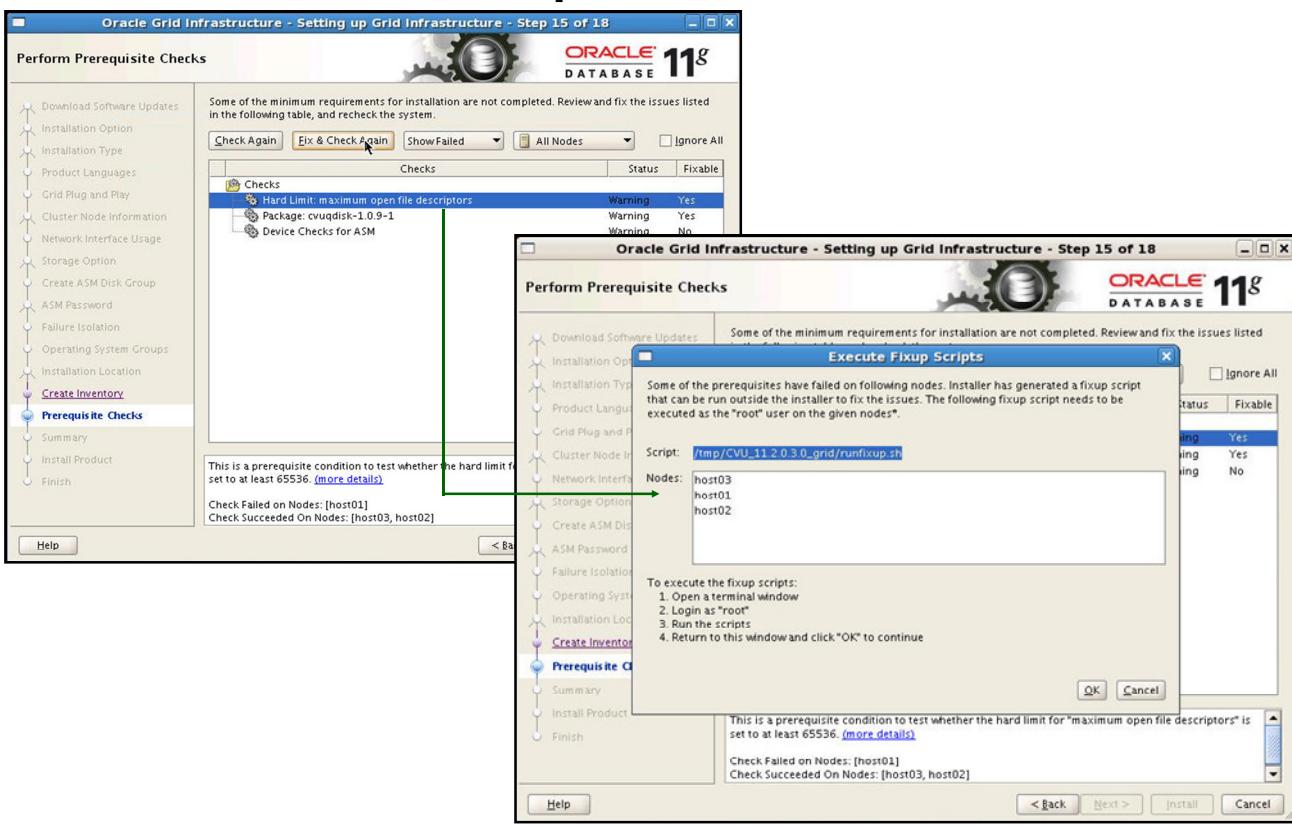
Next, OUI displays the Specify Installation Location page. Enter a base location for installing Oracle software. A separate Oracle base should be created for each operating system user owning an Oracle software installation. In the example in the slide, that location is /u01/app/grid.

In the Software Location field, enter a directory in which to install Oracle Grid Infrastructure. In the example, the grid operating system user will own this installation, so the software location should be /u01/app/11.2.0/grid. When you have entered proper Oracle Base and Software Location values, click Next to continue.

On the Create Inventory page, enter a location under your Oracle Base to store the Oracle Inventory. Click Next to continue.

Note: The Create Inventory page is common to both Advanced and Typical installation types as are the remaining installation steps.

Prerequisite Checks



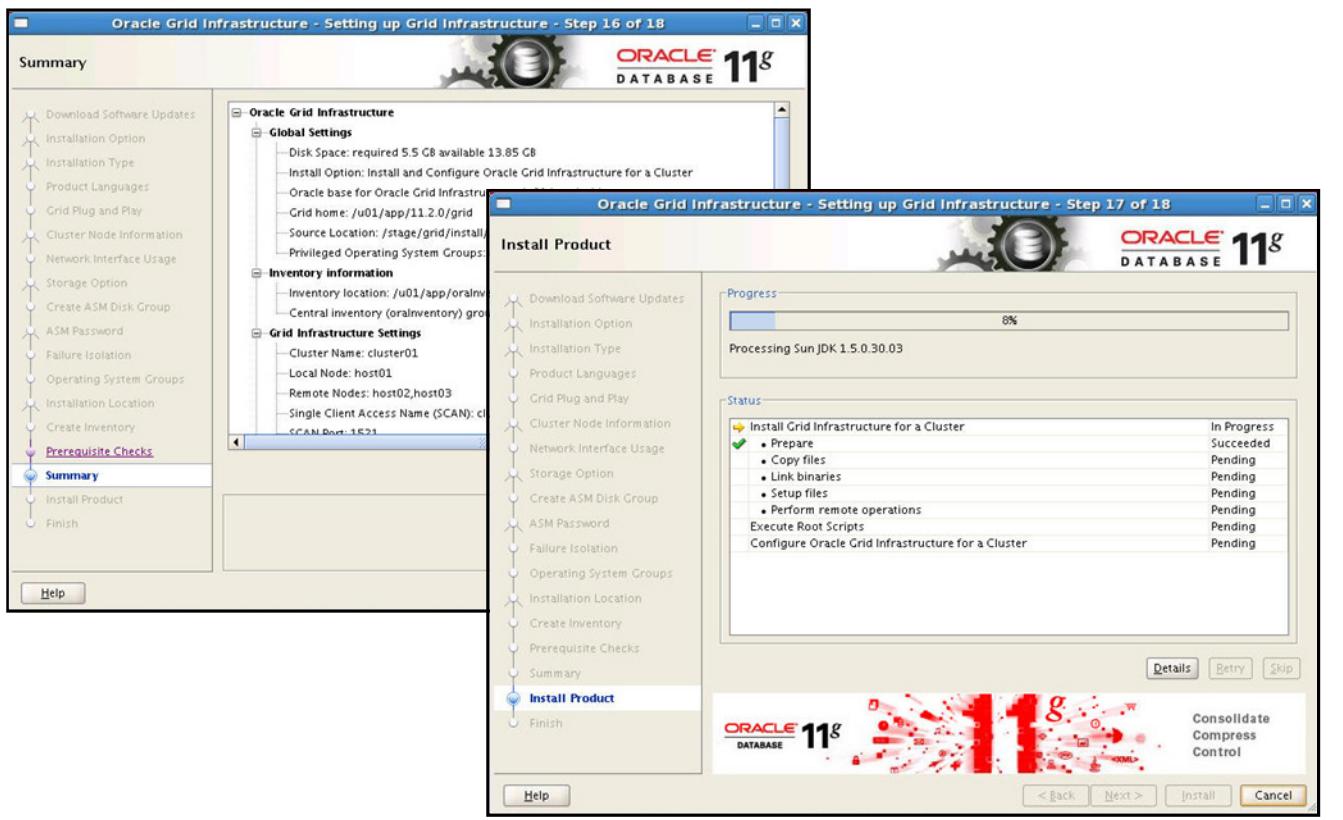
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

With the introduction of Oracle Database 11g Release 2, the OUI provides additional guidance to ensure recommended deployment and to prevent configuration issues. In addition, configuration assistants validate configurations and provide scripts to fix issues that you can use or choose to reject. Clicking the Fix & Check Again button opens a window instructing you to run a fixup script that must be run as `root` on all nodes before continuing.

The installer can determine whether the deficiency can be corrected, presenting the user performing the installation with the option of allowing the OUI to correct the situation. By clicking the Fix & Check Again button, a script is generated on the nodes where the deficient condition exists. After executing the scripts as `root` on the nodes, the kernel parameter is adjusted, allowing the installation to continue uninterrupted. When the prerequisite checks have completed successfully, click Next to continue. It is possible that an installation deficiency cannot be corrected with a generated fixup script. The installer will indicate this with a "No" in the Fixable column for that item.

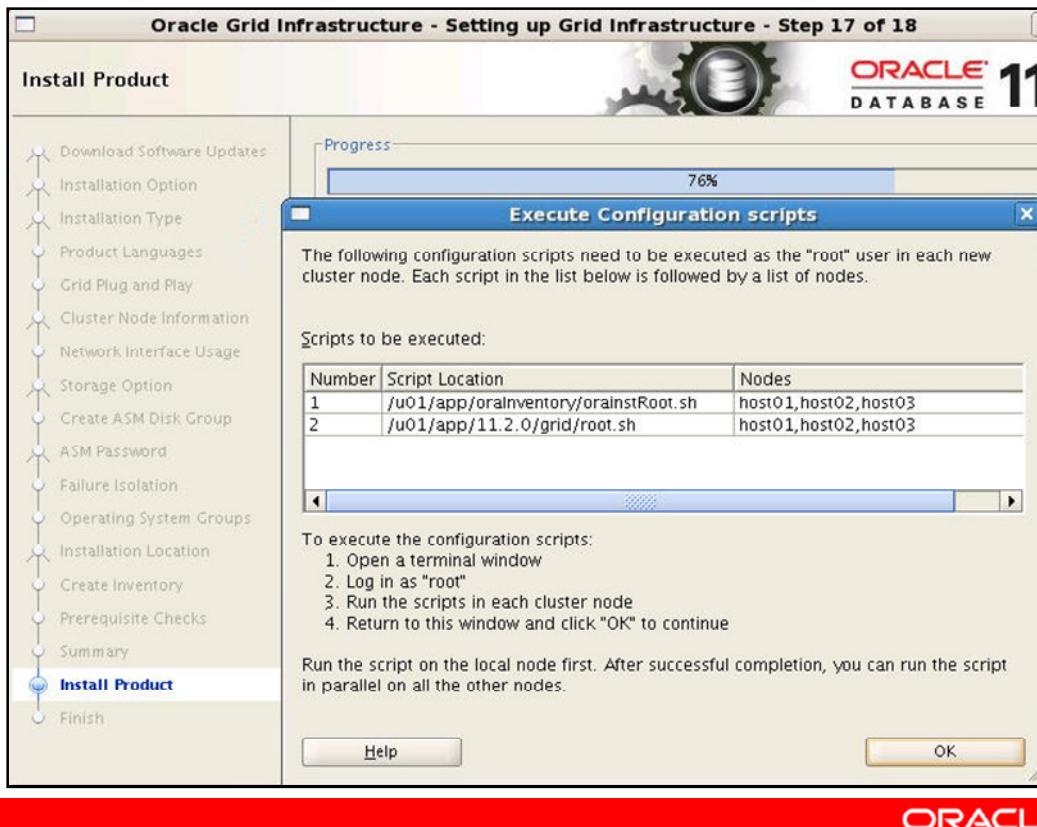
Finishing the Installation



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When the Summary page appears, check the installation information displayed there. If the information is correct, click the Finish button to begin the software setup. You can monitor the progress of the Grid Infrastructure installation on the Install Product screen.

Finishing the Installation



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

After the Grid Infrastructure has been copied to the remote nodes, you will be prompted to run root scripts on all installed nodes. If this is the first Oracle software initiated by the operating system user, you will run the `orainstRoot.sh` scripts on all nodes as prompted by the installer. Next, you must execute the `root.sh` script on all nodes. When the `root.sh` script finishes executing on the last node, click Finish to exit the installer.

Verifying the Grid Infrastructure Installation

```
[grid@host01 bin]$ ./crsctl stat res -t
NAME        TARGET  STATE     SERVER          STATE_DETAILS
-----
Local Resources
-----
ora.DATA.dg      ONLINE  ONLINE    host01
                  ONLINE  ONLINE    host02
                  ONLINE  ONLINE    host03
ora.LISTENER.lsnr
                  ONLINE  ONLINE    host01
                  ONLINE  ONLINE    host02
                  ONLINE  ONLINE    host03
ora.asm         ONLINE  ONLINE    host01          Started
                  ONLINE  ONLINE    host02          Started
                  ONLINE  ONLINE    host03          Started
ora.net1.network
                  ONLINE  ONLINE    host01
                  ONLINE  ONLINE    host02
                  ONLINE  ONLINE    host03
...
Cluster Resources
-----
ora.LISTENER_SCAN1.lsnr
  1      ONLINE  ONLINE    host02
ora.LISTENER_SCAN2.lsnr
  1      ONLINE  ONLINE    host03
ora.LISTENER_SCAN3.lsnr
  1      ONLINE  ONLINE    host01
ora.cvu           1      ONLINE  ONLINE    host01
ora.host01.vip   1      ONLINE  ONLINE    host01
...
ora.scan1.vip   1      ONLINE  ONLINE    host02
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Execute the `crsctl` command as shown in the slide to confirm that all cluster resources are up and running. If you elected to incorporate GNS in your Grid Infrastructure installation, you should confirm that your DNS is properly forwarding address requests for your application and SCAN VIPs to your GNS and that they are resolved properly. You can do this with `dig`.

Execute the `dig` command with DNS and VIP addresses as shown:

```
# dig @myDNS.example.com racnode01-cluster01.example.com
...
;; QUESTION SECTION:
;racnode01-vip.cluster01.example.com. IN A

;; ANSWER SECTION:
racnode01-vip.cluster01.example.com. 120 IN A 192.0.2.103
...
# dig @myDNS.example.com cluster01-scan.cluster01.example.com
...
;; ANSWER SECTION:
cluster01-scan.cluster01.example.com. 120 IN A 192.0.2.248
cluster01-scan.cluster01.example.com. 120 IN A 192.0.2.253
cluster01-scan.cluster01.example.com. 120 IN A 192.0.2.254
```

Modifying Oracle Clusterware Binaries After Installation

After installation, if you need to modify the Oracle Clusterware configuration, then you must unlock the Grid home.

1. Log in as root, go to `<Grid_home>/crs/install`, and unlock the Grid home using the following command:

```
# perl rootcrs.pl -unlock -crshome /u01/app/11.2.0/grid
```

2. As the Grid software owner, relink binaries. The example updates the interconnect protocol from UDP to IPC:

```
# su - grid  
$ make -f $ORACLE_HOME/rdbms/lib/ins_rdbms.mk ipc_rds ioracle
```

3. Relock the Grid home and restart the cluster using the following command: `# perl rootcrs.pl -patch`
4. Repeat steps 1 through 3 on each cluster member node.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

After installation, if you need to modify the Oracle Clusterware configuration, then you must unlock the Grid home. For example, if you want to apply a one-off patch, or if you want to modify an Oracle Exadata configuration to run IPC traffic over RDS on the interconnect instead of using the default UDP, then you must unlock the Grid home. Unlock the home using the following procedure:

1. Log in as root, and change directory to the path `Grid_home/crs/install`, where `Grid_home` is the path to the Grid home, and unlock the Grid home using the command `rootcrs.pl -unlock -crshome Grid_home`, where `Grid_home` is the path to your Grid infrastructure home. For example, with the Grid home `/u01/app/11.2.0/grid`, enter the following command:
`# cd /u01/app/11.2.0/grid/crs/install`
`# perl rootcrs.pl -unlock -crshome /u01/app/11.2.0/grid`
2. Change user to the Oracle Grid Infrastructure software owner and relink binaries using the command syntax `make -f Grid_home/rdbms/lib/ins_rdbms.mk target`, where `Grid_home` is the Grid home and `target` is the binaries that you want to relink. For example, where the grid user is `grid`, `$ORACLE_HOME` is set to the Grid home, and where you are updating the interconnect protocol from UDP to IPC, you enter the following command:
`# su grid`
`$ make -f $ORACLE_HOME/rdbms/lib/ins_rdbms.mk ipc_rds ioracle`

3. Relock the Grid home and restart the cluster using the following command:

```
# perl rootcrs.pl -patch
```
4. Repeat steps 1 through 3 on each cluster member node.

Module 3: Configuring ASM Disk Groups

A solid red horizontal bar spanning most of the page width.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Creating a New Disk Group

The CREATE DISKGROUP command creates new ASM disk groups.

```
CREATE DISKGROUP diskgroup_name
  [ { HIGH | NORMAL | EXTERNAL } REDUNDANCY ]
  { [ FAILGROUP failgroup_name ]
    DISK qualified_disk_clause [, qualified_disk_clause]...
  } ...
  [ ATTRIBUTE { 'attribute_name' = 'attribute_value' }... ]
;

qualified_disk_clause ::= search_string
[ NAME disk_name ]
[ SIZE size_clause ]
[ FORCE | NOFORCE ]
```

- Example:

```
CREATE DISKGROUP FRA NORMAL REDUNDANCY
DISK 'ORCL:SDD11' NAME 'FRA_DISK1' SIZE 977 M,
      'ORCL:SDD12' NAME 'FRA_DISK2' SIZE 977 M;
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The CREATE DISKGROUP statement creates a disk group, assigns one or more disks to the disk group, and mounts the disk group for the first time. If you want ASM to automatically mount the disk group when an ASM instance starts, you must add the disk group name to the value of the `ASM_DISKGROU`PS initialization parameter in your parameter files (PFILEs). If you use a server parameter file (SPFILE), the disk group is added to the initialization parameter automatically.

The CREATE DISKGROUP command can also be run using database management tools such as the ASM Configuration Assistant (ASMCA), Oracle Enterprise Manager, and ASM Command-Line utility (ASMCMD).

The CREATE DISKGROUP statement has the following clauses:

- REDUNDANCY clause

The REDUNDANCY clause allows you to specify the redundancy level of the disk group.

`NORMAL REDUNDANCY` requires the existence of at least two failure groups. By default, `NORMAL REDUNDANCY` provides a two-way mirror of all ASM files except for control files, which are mirrored three ways. `NORMAL REDUNDANCY` disk groups can tolerate the loss of one failure group.

HIGH REDUNDANCY requires the existence of at least three failure groups. ASM fixes mirroring at three-way mirroring, with each file getting two mirrored copies. HIGH REDUNDANCY disk groups can tolerate the loss of two failure groups.

EXTERNAL REDUNDANCY indicates that ASM does not provide any redundancy for the disk group. The disks within the disk group must provide redundancy (for example, using a storage array), or you must be willing to tolerate the loss of the disk group if a disk fails. You cannot specify the FAILGROUP clause if you specify EXTERNAL REDUNDANCY.

- FAILGROUP clause

Use this clause to specify a name for one or more failure groups. If you omit this clause, and you have specified NORMAL or HIGH REDUNDANCY, then ASM automatically adds each disk in the disk group to its own failure group. The implicit name of the failure group is the same as the name in the NAME clause.

- DISK clause

Use this clause to specify one or more disks for each failure group.

For each disk that you are adding to the disk group, specify the operating system-dependent search string that ASM will use to find the disk. The `search_string` must point to a subset of the disks returned by discovery using the strings in the `ASM_DISKSTRING` initialization parameter. If `search_string` does not point to any disks to which the ASM user has read/write access, then ASM returns an error. If it points to one or more disks that have already been assigned to a different disk group, then Oracle Database returns an error unless you also specify FORCE. For each valid candidate disk, ASM formats the disk header to indicate that it is a member of the new disk group.

The optional NAME subclause is valid only if the `search_string` points to a single disk. It specifies an operating system-independent name for the disk. The name can be up to 30 alphanumeric characters. The first character must be alphabetic. If you omit this clause and you assigned a label to a disk through ASMLib, then that label is used as the disk name. If you are not using ASMLib, then ASM creates a default name of the form `diskgroup_name_nnnn`, where `nnnn` is the disk number. You can use this name to refer to the disk in subsequent ASM operations.

Use the optional SIZE subclause to specify the size of the disk. If you specify a size greater than the capacity of the disk, then ASM returns an error. If you specify a size less than the capacity of the disk, you limit the disk space ASM will use. If you omit this clause, ASM attempts to determine the size of the disk programmatically.

You can specify FORCE or NOFORCE for each disk.

Specify FORCE if you want ASM to add the disk to the disk group even if the disk is already a member of a different disk group. Exercise caution because using FORCE in this way may destroy existing disk groups. For this clause to be valid, the disk must already be a member of a disk group and the disk cannot be part of a mounted disk group.

Specify NOFORCE if you want ASM to return an error if the disk is already a member of a different disk group. NOFORCE is the default.

- ATTRIBUTE clause

Use this clause to set attribute values for the disk group. ASM disk group attributes are described later in this lesson.

Creating a New Disk Group with ASMCMD

ASMCMD allows an XML configuration file to create and change the disk group.

- Sample XML used with the `mkdg` command:

```
<dg name="DATA" redundancy="normal">
    <fg name="fg1">
        <dsk string="/dev/sda1" />
        <dsk string="/dev/sdb1" />
    </fg>
    <fg name="fg2">
        <dsk string="/dev/sdc1" />
        <dsk string="/dev/sdd1" />
    </fg>
    <a name="compatible.asm" value="11.2"/>
    <a name="compatible.rdbms" value="11.2"/>
</dg>
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ASMCMD has added the ability to use an XML configuration file to either create a disk group or change a disk group configuration.

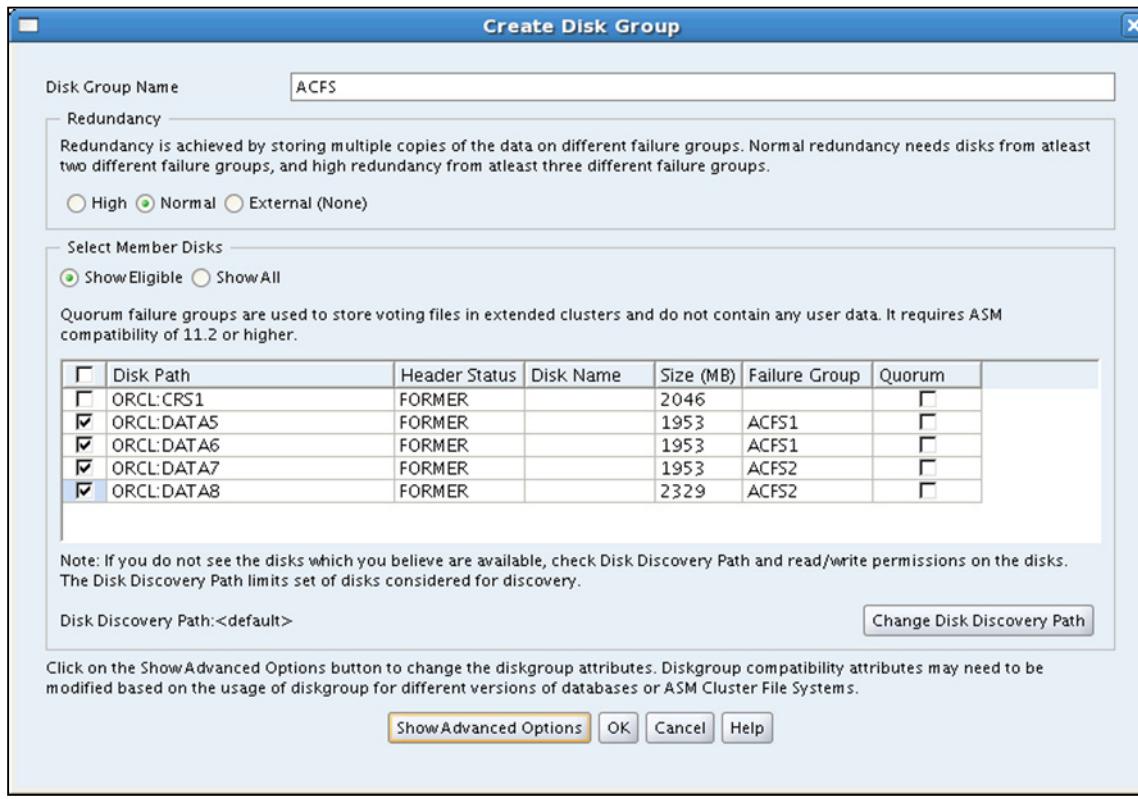
The XML file for the `mkdg` command specifies the name of the disk group, redundancy, attributes, and paths of the disks that form the disk group. Redundancy is an optional parameter; the default is normal redundancy. For some types of redundancy, disks are required to be gathered into failure groups. In the case that failure groups are not specified, every disk will be in its own failure group.

It is possible to set the disk group attribute values during disk group creation. Some attributes, such as `AU_SIZE` and `SECTOR_SIZE`, can be set only during disk group creation.

The following is an example of an inline XML configuration for `chdg`. This XML alters the disk group named `DATA`. The `FG1` failure group is dropped and the `DATA_0001` disk is also dropped. The `/dev/disk5` disk is added to the `FG2` failure group. The rebalance power level is set to 3.

```
ASMCMD> chdg '<chdg> <dg name="DATA" power="3"> <drop> <fg name="FG1"> </fg> <dsk name="DATA_0001" /> </drop> <add> <fg name="FG2"> <dsk string="/dev/disk5"/> </fg> </add> </chdg>'
```

Creating an ASM Disk Group with ASMCA



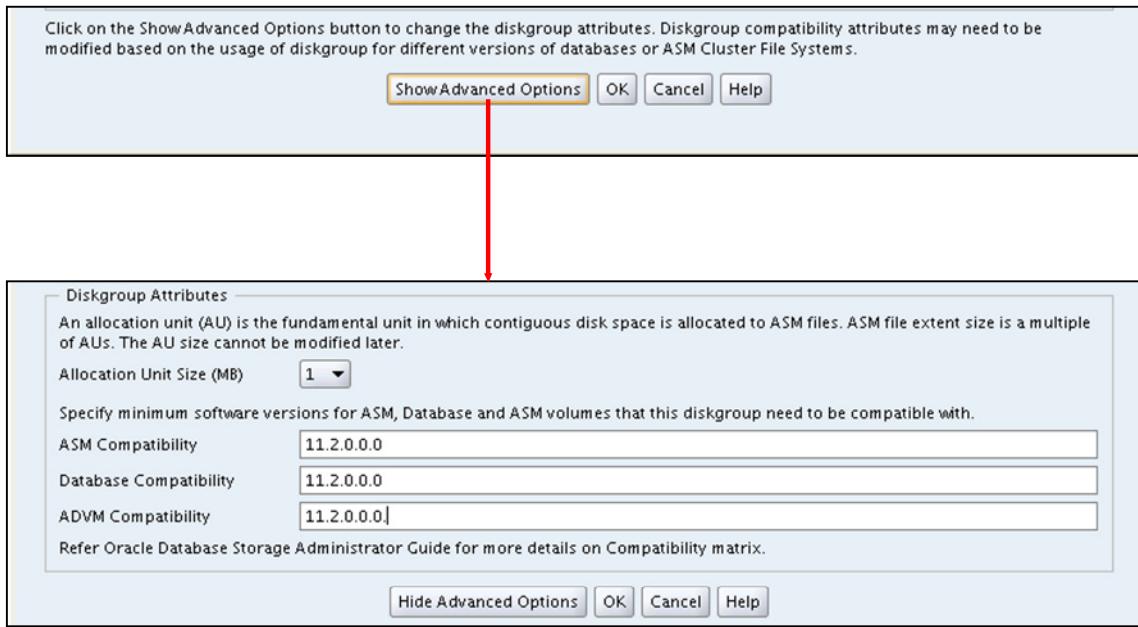
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the slide, the ASMCA utility is being used to create a disk group. Redundancy is set to Normal. Four disks are selected to be members of this disk group: ORCL : DATA5, ORCL : DATA6, ORCL : DATA7, and ORCL : DATA8. The first two disks are placed in the failure group ACFS1, and the second two disks are in the failure group ACFS2. The header status can show multiple valid values. When this is the first time a disk has been created, the header status will be CANDIDATE. If the disk has been a member of the disk group and has been cleanly dropped, the header status will be FORMER as shown in the slide. The PROVISIONED header status is similar to CANDIDATE except that PROVISIONED implies that an additional platform-specific action has been taken by an administrator to make the disk available for ASM.

The disk discovery path is set by an initialization parameter, `ASM_DISKSTRING`. This parameter can consist of multiple search paths separated by commas. The default `ASM_DISKSTRING` parameter string is empty. There is an appropriate default discovery path for most OS platforms. A disk discovery string that limits the directories that are searched can reduce the time for discovery, but the disk discovery string must include all disks that are members of existing disk groups. The default string will allow ASM to find disks that have been initialized with the Oracle ASMLib utility, `oracleasm`. Advanced options for disk groups are shown later in this lesson.

Creating an ASM Disk Group: Advanced Options



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

At the bottom of the Create Disk Group page, click Show Advanced Options to see the portion of the page shown in the slide. On this portion of the page, you can set disk group attributes: allocation unit size and compatibility parameters.

The field labeled ASM Compatibility sets the `COMPATIBLE.ASM` attribute. For Oracle ASM in Oracle Database 11g, 10.1 is the default setting for the `COMPATIBLE_ASM` attribute when using the SQL `CREATE DISKGROUP` statement, the `ASMCMD mkdg` command, and the Oracle Enterprise Manager Create Disk Group page. When creating a disk group with ASMCA, the default setting is 11.2.0. When setting the values for the compatibility attributes, specify at least the major and minor versions of a valid Oracle Database release number. For example, you can specify compatibility as "11.1" or "11.2"; Oracle assumes that any missing version number digits are zeroes.

The Database Compatibility sets the minimum version level for any database instance that is allowed to use the disk group. This is the `COMPATIBLE.RDBMS` attribute.

To use ADVM volumes, ADVM Compatibility must be set to 11.2.0 or higher and the ASM Compatibility must be 11.2.0 or higher. The ADVM Compatibility field sets the `COMPATIBLE.ADVM` attribute.

Note: Advancing the values for disk group compatibility attributes is an irreversible operation.

Creating a Disk Group with Enterprise Manager

The screenshot shows the Oracle Enterprise Manager 11g Database Control interface. The title bar reads "ORACLE Enterprise Manager 11g Database Control". The top menu includes "Setup", "Preferences", "Help", "Logout", "Cluster", and "Database". The user is logged in as "SYS / SYSASM". The main title is "Automatic Storage Management: +ASM1_host01.example.com". Below it, there are tabs for "Home", "Performance", "Disk Groups" (which is selected), "Configuration", "Users", and "ASM Cluster File System". A toolbar below the tabs includes "Create", "Mount All", and "Dismount All" buttons, along with "Mount", "Dismount", "Rebalance", "Check", and "Delete" links. A "Select All" or "Select None" checkbox is present. A table displays disk group information:

Select	Name	State	Redundancy	Size (GB)	Used (GB)	Used (%)	Usable Free (GB)	Member Disks
<input type="checkbox"/>	DATA	MOUNTED	NORMAL	8.00	4.55	56.95	1.11	4

Below the table are two tips:

- TIP** The usable free space specifies the amount of space that can be safely used for data. A value above zero means that redundancy can be properly restored after a disk failure.
- TIP** Mount All and Dismount All operation will only mount and dismount the disk groups specified in the Auto Mount Disk Groups parameter.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The same functionality is available with Enterprise Manager. This slide shows the Disk Group page of the Automatic Storage Management target.

Note: Enterprise Manager is not part of the Grid Infrastructure installation. You must install a database instance to have access to Enterprise Manager Database Control that is shown in the slide.

Creating a Disk Group with Enterprise Manager

Automatic Storage Management: +ASM1_host01.example.com

Home Performance **Disk Groups** Configuration Users ASM Cluster File System

Create Mount All Dismount All

Create Disk Group

Name: ACFS
Redundancy: NORMAL
Allocation Unit (MB): 1

An allocation unit (AU) is the fundamental unit in which contiguous disk space is allocated to ASM files. ASM file extent size is a multiple of AUs. The AU size cannot be modified later.

Mount this disk group on all Automatic Storage Management instances in this cluster

Candidate Member Disks

Select	Path	Header Status	Library	Label	ASM Disk Name	Size	Unit	Force Reuse	Failure Group	Quorum
<input type="checkbox"/>	ORCL:CRS1	FORMER	ASM LIBRARY - GENERIC LINUX, VERSION 2.0.4 (KABI_V2)	CRS1		2046	MB	<input type="checkbox"/>		<input type="checkbox"/>
<input checked="" type="checkbox"/>	ORCL:DATA5	FORMER	ASM LIBRARY - GENERIC LINUX, VERSION 2.0.4 (KABI_V2)	DATA5		1953	MB	<input type="checkbox"/>	ACFS1	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ORCL:DATA6	FORMER	ASM LIBRARY - GENERIC LINUX, VERSION 2.0.4 (KABI_V2)	DATA6		1953	MB	<input type="checkbox"/>	ACFS1	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ORCL:DATA7	FORMER	ASM LIBRARY - GENERIC LINUX, VERSION 2.0.4 (KABI_V2)	DATA7		1953	MB	<input type="checkbox"/>	ACFS2	<input type="checkbox"/>
<input checked="" type="checkbox"/>	ORCL:DATA8	FORMER	ASM LIBRARY - GENERIC LINUX, VERSION 2.0.4 (KABI_V2)	DATA8		2329	MB	<input type="checkbox"/>	ACFS2	<input type="checkbox"/>

TIP Quorum failure groups are used to store voting files in extended clusters and do not contain any user data. It requires ASM compatibility of 11.2 or higher.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Just as with the ASMCA utility, you can specify the disks and failure groups on the Create Disk Group page in Enterprise Manager. To change the disk discovery string, click the Configuration tab. The Compatibility Options are available on the Create Disk Group page, but that section is not shown in the slide.

Summary

In this lesson, you should have learned how to:

- Perform preinstallation tasks for Grid Infrastructure
- Install Grid Infrastructure
- Verify the installation
- Configure ASM disk groups



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

15

Administering Oracle Clusterware

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Display Clusterware management proficiency
- Demonstrate Oracle Cluster Registry (OCR) backup and recovery techniques



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Managing Oracle Clusterware

- Command-line utilities
 - `crsctl` manages clusterware-related operations:
 - Starting and stopping Oracle Clusterware
 - Enabling and disabling Oracle Clusterware daemons
 - Registering cluster resources
 - `srvctl` manages Oracle resource-related operations:
 - Starting and stopping database instances and services



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Ongoing management of Oracle Clusterware is achieved by using the `crsctl` and `srvctl` command-line utilities installed under the Oracle Grid Infrastructure home directory.

Oracle Clusterware components and resources can be monitored and managed from any node in the cluster by using `crsctl`. The `srvctl` utility provides similar monitoring and management capabilities for Oracle-related resources such as database instances and database services. Both utilities are provided with Oracle Clusterware. However, most `crsctl` commands are available only to clusterware administrators, whereas `srvctl` commands are available to other groups such as database administrators.

Controlling Oracle Clusterware

The `crsctl` utility can be used to control Oracle Clusterware.

- To start or stop Oracle Clusterware on a specific node:

```
# crsctl start crs
```

```
# crsctl stop crs
```

- To enable or disable Oracle Clusterware on a specific node:

```
# crsctl enable crs
```

```
# crsctl disable crs
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When a node that contains Oracle Clusterware is started, the Oracle Clusterware wrapper script is automatically started by the `/etc/init.d/ohasd` startup script. When the `crsctl` utility is used to disable Cluster Ready Services (CRS) from automatically starting, state information related to startup is placed in the `SLCS_SRC` control files, preventing automatic startup on machine reboot. To check the status of CRS, use the following syntax:

```
# crsctl check crs
CRS-4638: Oracle High Availability Services is online
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
```

You may have to manually control the Oracle Clusterware stack while applying patches or during planned outages. You can stop Oracle Clusterware by using the `crsctl stop crs` command and start it by using the `crsctl start crs` command.

Verifying the Status of Oracle Clusterware

The `crsctl` utility can be used to verify the status of Oracle Clusterware.

- To determine the overall health on a specific node:

```
$ crsctl check crs
CRS-4638: Oracle High Availability Services is online
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
```

- To check the viability of Cluster Synchronization Services (CSS) across nodes:

```
$ crsctl check cluster
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `crsctl` utility can be used to verify the status of Oracle Clusterware on specific nodes and across nodes. In contrast to the `crsctl` controlling commands that required the `root` access (shown in the previous slide), the `check` commands do not require `root` and may be executed by the Oracle Clusterware software owner. The overall health of the clusterware on a specific node can be obtained by using the `crsctl check crs` command. It is possible to target three of the individual daemons by using the `crsctl check <daemon>` command for the `crsd`, `evmd`, and `cssd` daemons only. These commands are processed only on the node on which they are executed. To check the viability of Cluster Synchronization Services (CSS) across all nodes, use the `crsctl check cluster` command. The output of the overall health check performed on a specific node is shown in the slide.

Determining the Location of Oracle Clusterware Configuration Files

The two primary configuration file types for Oracle Clusterware are the voting disk and the Oracle Cluster Registry (OCR).

- To determine the location of the voting disk:

```
# crsctl query css votedisk
##  STATE  File Universal Id                               File Name  Disk group
--  -----
1. ONLINE  8c2e45d734c64f8abf9f136990f3daf8  (ASMDISK01) [DATA]
2. ONLINE  99bc153df3b84fb4bf071d916089fd4a  (ASMDISK02) [DATA]
3. ONLINE  0b090b6b19154fc1bf5913bc70340921  (ASMDISK03) [DATA]

Located 3 voting disk(s).
```

- To determine the location of the OCR:

```
$ cat /etc/oracle/ocr.loc
ocrconfig_loc=+DATA
local_only=FALSE
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Clusterware uses two primary configuration file types: the voting disk and the Oracle Cluster Registry (OCR). There can be multiple redundant copies of each. You can determine the location of the voting disk by using the `crsctl query css votedisk` command on any node. This does not require the CSS daemons to be running, and the command can be executed as the Grid Infrastructure owner. The location of the OCR file can be determined by using the `cat /etc/oracle/ocr.loc` command. Because these files are always located on shared storage, the command can be executed from any node.

Note: The OCR can also be located by using the `ocrcheck` utility, provided that the path to the utility is known or the path has been added to the `PATH` environment variable.

Checking the Integrity of Oracle Clusterware Configuration Files

The following techniques are used to validate the integrity of Oracle Cluster configuration files.

- Check the `ocssd.log` for voting disks issues.

```
$ grep voting <grid_home>/log/<hostname>/cssd/ocssd.log
```

- Use the `cluvfy` utility or the `ocrcheck` command to check the integrity of the OCR.

```
$ cluvfy comp ocr -n all -verbose
```

```
$ ocrcheck
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To check the integrity of the voting disks, examine `ocssd.log`. Errors with the voting disks appear in the log. The following is a snippet of the output that indicates what an error may look like:

```
$ grep voting ocssd.log
[    CSSD]2008-09-09 10:47:09.711 [100494224] >ERROR:
  clssnmvReadFatal: voting device corrupt
  (0x00000000/0x00000000/1//dev/sda6)
[    CSSD]2008-09-09 10:47:09.711 [3082128272] >ERROR:
  clssnmvReadFatal: voting device corrupt
  (0x00000000/0x00000000/2//dev/sda7)
```

Two commands may be used to check the integrity of the OCR file. They are:

```
$ ocrcheck
$ cluvfy comp ocr -n all -verbose
```

Backing Up and Recovering the Voting Disk

- In Oracle Clusterware 11g Release 2, voting disk data is backed up automatically in the OCR as part of any configuration change.
- Voting disk data is automatically restored to any added voting disks.
- Using `dd` to back up and restore a voting disk ***may result in the loss of the voting disk!***
- To add or remove voting disks on non-Automatic Storage Management (ASM) storage, use the following commands:

```
# crsctl delete css votedisk path_to_voting_disk  
# crsctl add css votedisk path_to_voting_disk
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Backing Up the Voting Disk

In previous releases, backing up the voting disks using a `dd` command was a required postinstallation task. With Oracle Clusterware 11g Release 2, backing up and restoring a voting disk using the `dd` command may result in the loss of the voting disk, so this procedure is not supported.

Backing up voting disks manually is no longer required because voting disk data is backed up automatically in the OCR as part of any configuration change and voting disk data is automatically restored to any added voting disks.

Recovering Voting Disks

If you have multiple voting disks on non-ASM storage, you can remove the voting disks and add them back into your environment with all the information from the other voting disks using the following commands, where `path` is the complete path of the location where the voting disk resides:

```
crsctl delete css votedisk path_to_voting_disk  
crsctl add css votedisk path_to_voting_disk
```

Note: You can migrate voting disks from non-ASM storage options to ASM without taking down the cluster. To use an ASM disk group to manage the voting disks, you must set the `COMPATIBLE_ASM` attribute to 11.2.0.0.

Adding, Deleting, or Migrating Voting Disks

- To add or delete one or more voting disks to non-ASM storage:

```
# crsctl add css votedisk path_to_new_voting_disk  
# crsctl delete css votedisk path_to_old_voting_disk
```

- To add a voting disk to ASM:

```
# crsctl replace votedisk +asm_disk_group
```

- To migrate voting disks from non-ASM storage devices to ASM or vice versa, specify the ASM disk group name or path to the non-ASM storage device:

```
# crsctl replace votedisk {+asm_disk_group |  
path_to_voting_disk}
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To add one or more voting disks to non-ASM storage, run the following command as root:

```
# crsctl add css votedisk path_to_voting_disk [...]
```

To add a voting disk to ASM:

```
# crsctl replace votedisk +asm_disk_group
```

To replace voting disk A with voting disk B on non-ASM storage, first add voting disk B and then delete voting disk A:

```
# crsctl add css votedisk path_to_voting_diskB  
# crsctl delete css votedisk path_to_voting_diskA
```

Use the crsctl replace votedisk command to replace a voting disk on ASM. You do not have to delete any voting disks from ASM using this command.

To remove a voting disk, run the following command as root, replacing the *voting_disk_GUID* variable with one or more space-delimited, voting disk globally unique identifiers (GUIDs) you want to remove:

```
# crsctl delete css votedisk voting_disk_GUID
```

To migrate voting disks from non-ASM storage devices to ASM or vice versa, specify the ASM disk group name or path to the non-ASM storage device in the following command:

```
# crsctl replace votedisk {+asm_disk_group | path_to_voting_disk}
```

You can run this command on any node in the cluster.

Locating the OCR Automatic Backups

- The OCR is backed up automatically.
- Only one node performs the backup.
- To determine the node and location of the backup:

```
$ ocrconfig -showbackup auto
host02 2009/07/28 12:20:42 /u01/app/.../cdata/cluster01/backup00.ocr
host02 2009/07/28 08:20:41 /u01/app/.../cdata/cluster01/backup01.ocr
host02 2009/07/28 04:20:40 /u01/app/.../cdata/cluster01/backup02.ocr
host02 2009/07/27 16:20:37 /u01/app/.../cdata/cluster01/day.ocr
host02 2009/07/28 00:20:39 /u01/app/.../cdata/cluster01/week.ocr
```

- Files could be spread across nodes due to outages.
- The backup frequency and retention policies are:
 - Every four hours: CRS keeps the last three copies.
 - At the end of every day: CRS keeps the last two copies.
 - At the end of every week: CRS keeps the last two copies.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The information contained in the OCR is much more dynamic in nature than the voting disk. Oracle Clusterware automatically performs routine backups of the OCR file. These are physical backups. Only one node has the responsibility to perform these backups, but that responsibility can transfer to any other node in the cluster when outages occur. The default target location of each automatically generated OCR backup file is the *<Grid Home>/cdata/<cluster name>* directory.

The automatic backup is on a four-hour schedule, but only a limited number of files are retained. Only the last three backups of the routine four-hour intervals are kept, with newer backups overwriting older ones. At the end of the day, a backup is taken and the last two are retained. At the end of the week, a backup is taken and the last two are retained. In conclusion, there should not be more than seven automatic backups that require storage: one four-hours old, one eight-hours old, one 12-hours old, one 24-hours old, one 48-hours old, one seven-days old, and one 14-days old. The four-hour backup interval is not based on the time of the day, but instead on an offset from the time that the clusterware was started.

The backup file names cannot be changed and are named as follows: `backup00.ocr`, `backup01.ocr`, `backup02.ocr`, `day.ocr`, `day_.ocr`, `week.ocr`, and `week_.ocr`.

Changing the Automatic OCR Backup Location

- The automatic backup location should be changed to a location shared by all nodes.

```
# ocrconfig -backuploc <path to shared CFS or NFS>
```

- The backup location will be used for both automatic and manual backups.
- It is recommended that these files be included in routine scheduled backups to an offline location.
- If CRS has been stopped on all nodes, the schedule of backups is suspended.
- On restart, a backup is not immediately taken and the backup timer is reset.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Because the automatic backup is performed only by the master node to the local file system by default, it is recommended that you change the OCR automatic backup location to one that is shared by all nodes in the cluster by using the `ocrconfig -backuploc <new location>` command. This command takes one argument that is the full path of the directory name for the new location. The location will be used for both automatic and manual backups. You cannot customize the backup frequencies, the number of retained copies, or the names of the backup files. If CRS on the master node is shut down, another node becomes the master, and backups will resume on that node. If the backup location has not been changed to a common shared location, backups will exist locally across multiple nodes potentially. If CRS is stopped on all nodes during a scheduled backup, on restart, a backup will not be immediately taken and the backup timer will be reset. This could result in a longer time duration between automatic backups than the standard four-hour interval.

Because of the importance of the OCR information, it is also recommended that you manually create copies of automatically generated physical backups. You can use any backup software to copy the automatically generated backup files, and it is recommended that you do that at least once daily to a different device from where the automatic backups are.

Note: Do not place your automatic OCR backups on ASM Cluster File System (ACFS) storage.

Adding, Replacing, and Repairing OCR Locations

- Add an OCR location to either ASM or other storage device:

```
# ocrconfig -add +DATA2  
# ocrconfig -add /dev/sde1
```

- To replace the current OCR location:

```
# ocrconfig -replace /dev/sde1 -replacement +DATA2
```

- To repair OCR configuration, run this command on the node on which you have stopped Oracle Clusterware:

```
[root@host03]# ocrconfig -repair -add +DATA1
```

You cannot perform this operation on a node on which Oracle Clusterware is running.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can add an OCR location after an upgrade or after completing an Oracle Grid Infrastructure installation. Oracle Clusterware can manage up to five redundant OCR locations. As root, run the following command to add an OCR location to either ASM or other storage device:

```
# ocrconfig -add +asm_disk_group | file_name
```

To replace the current OCR location using either *destination_file* or *+ASM_disk_group* to indicate the current and target OCR locations:

```
# ocrconfig -replace destination_file | +ASM_disk_group -replacement destination_file | +ASM_disk_group
```

It may be necessary to repair an OCR configuration if your configuration changes while a node is stopped. Repairing an OCR configuration involves adding, deleting, or replacing an OCR location. To repair an OCR configuration, run *ocrconfig* on the node on which you have stopped Oracle Clusterware as root:

```
# ocrconfig -repair -add file_name | -delete file_name | -replace current_file_name -replacement new_file_name
```

This operation changes the OCR configuration only on the node on which you run this command. For example, if the OCR location is */dev/sde1*, use the command syntax *ocrconfig -repair -add /dev/sde1* on this node to repair its OCR configuration.

Removing an Oracle Cluster Registry Location

- To remove an OCR location, at least one other OCR must be online.
- Run the following command on any node in the cluster to remove an OCR location from either ASM or another shared location:

```
# ocrconfig -delete +DATA2  
# ocrconfig -delete /dev/sde1
```

- ***Do not*** perform an OCR removal unless there is at least one other active OCR location online, or you will get an error.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To remove an OCR location, at least one other OCR must be online. You can remove an OCR location to reduce OCR-related overhead or to stop mirroring your OCR because you moved OCR to redundant storage such as RAID.

Perform the following procedure as the `root` user to remove an OCR location from your Oracle Clusterware environment:

1. Ensure that at least one OCR location other than the OCR location that you are removing is online.
2. Run the following command on any node in the cluster to remove an OCR location from either ASM or another location:

```
# ocrconfig -delete +ASM_disk_group | file_name
```

The `file_name` variable can be a device name or a file name. This command updates the OCR configuration on all the nodes on which Oracle Clusterware is running.

Caution: Do not attempt to perform an OCR removal unless there is at least one other active OCR location online otherwise you will get an error. You cannot remove the last OCR file.

Migrating OCR Locations to ASM

1. Ensure that Oracle Clusterware is upgraded to 11g Release 2.

```
$ crsctl query crs activeversion  
Oracle Clusterware active version on cluster is [11.2.0.1.0]
```

2. Start ASM on all nodes and create a disk group that has at least 1 GB of space and has at least normal redundancy.
3. To add an OCR location to an ASM disk group, run the following command as root:

```
# ocrconfig -add +DATA2
```

4. To remove storage configurations no longer in use, run the following command as root:

```
# ocrconfig -delete /dev/raw/raw1  
# ocrconfig -delete /dev/raw/raw2
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To improve Oracle Clusterware storage manageability, OCR is configured, by default, to use ASM in Oracle Database 11g Release 2. With the Oracle Clusterware storage residing in an ASM disk group, you can manage both database and clusterware storage using EM.

However, if you upgrade from a previous version of Oracle Database, you can migrate your OCR location or locations to reside on ASM, and take advantage of the improvements in managing Oracle Clusterware storage. To migrate OCR locations to ASM using `ocrconfig`:

1. Ensure that Oracle Clusterware upgrade to 11g Release 2 is complete. Run the following command to verify the current running version:

```
$ crsctl query crs activeversion
```
2. Use ASM Configuration Assistant (ASMCA) to configure and start ASM on all nodes in the cluster, and then create a disk group that has at least 1 GB of space and has at least normal redundancy.
3. To add an OCR location to an ASM disk group, ensure that the Clusterware stack is running and run the following command as root:

```
# ocrconfig -add +new_disk_group
```

You can run this command more than once if you add more than one OCR location.
4. To remove storage configurations no longer in use, run the following command as root:

```
# ocrconfig -delete old_storage_location
```

Note: OCR inherits the redundancy of the disk group. If you want high redundancy for OCR, you must configure the disk group with high redundancy when you create it.

Migrating OCR from ASM to Other Shared Storage

1. Ensure that Oracle Clusterware is upgraded to 11g Release 2.

```
$ crsctl query crs activeversion  
Oracle Clusterware active version on cluster is [11.2.0.1.0]
```

2. Create at least one shared file with the following permissions: `root, oinstall, 640` making sure that the mount partition has at least 300 MB of space.
3. To add an OCR location, ensure that the Clusterware stack is running and run the following command as `root`:

```
# ocrconfig -add /dev/sde1  
# ocrconfig -add /dev/sdf1
```

4. To remove storage configurations no longer in use, run the following command as `root`:

```
# ocrconfig -delete +DATA2
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To migrate Oracle Clusterware storage from ASM to another storage choice:

1. Ensure that Oracle Clusterware upgrade to 11g Release 2 is complete. Run the following command to verify the current running version: `$ crsctl query crs activeversion`.
2. Create a shared file with the following permissions: `root, oinstall, 640`, making sure that the mount partition has at least 300 MB of space.
3. To add the file as an OCR location, ensure that the Oracle Clusterware stack is running and run the following command as `root`:

```
# ocrconfig -add new_file_location
```

You can run this command more than once if you add more than one OCR location.

4. To remove storage configurations no longer in use, run the following command as `root`:

```
# ocrconfig -delete old_storage_location
```

You can run this command more than once if there are multiple OCR locations configured.

Performing Manual OCR Backups

When significant changes to the configuration have occurred, a manual, on-demand backup is suggested.

- To perform a physical backup:

```
# ocrconfig -manualbackup
```

- To display a list of manual backups:

```
$ ocrconfig -showbackup manual
host02 2009/07/28 16:59:17
/u01/app/.../cdata/cluster01/backup_20090728_165917.ocr
```

- To perform a logical backup:

```
# ocrconfig -export /home/oracle/ocr.backup
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Unlike the voting disk, the OCR content can be very dynamic in nature, especially with the High Availability framework. If a significant amount of work has been done that would cause modifications to the OCR, it is recommended that a manual backup or export be performed before the routine automatic backup occurs. This on-demand backup can be used to restore the information if the OCR becomes corrupted or lost before the automatic backup occurs.

You are not allowed to specify the name used for the manual backup. A file will be created with the name `backup_<date>_<time>.ocr` and placed into the default backup location. When a manual backup is performed, it does not affect the automatic backup interval. The `export` command will create a binary file containing a logical backup of the OCR keys and values.

Most configuration changes that you make not only change the OCR contents but also cause file and database object creation. Some of these changes are often not restored when you restore the OCR. Do not perform an OCR restore as a correction to revert to previous configurations if some of these configuration changes fail. This may result in an OCR with contents that do not match the state of the rest of your system.

Recovering the OCR by Using Physical Backups

1. Locate a physical backup:

```
$ ocrconfig -showbackup
```

2. Stop the Oracle Clusterware stack on all nodes:

```
# crsctl stop cluster -all
```

3. Stop Oracle High Availability Services on all nodes:

```
# crsctl stop crs
```

4. Restore the physical OCR backup:

```
# ocrconfig -restore /u01/app/.../cdata/cluster01/day.ocr
```

5. Restart Oracle High Availability Services on all nodes:

```
# crsctl start crs
```

6. Check the OCR integrity:

```
$ cluvfy comp ocr -n all
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the following procedure to restore the OCR on UNIX-based systems:

1. Identify the OCR backups by using the `ocrconfig -showbackup` command. You can execute this command from any node as the `oracle` user. The output tells you on which node and which path to retrieve both automatically and manually generated backups. Use the `auto` or `manual` argument to display only one category.
2. Stop the Oracle Clusterware stack on all nodes by using the `crsctl stop cluster -all` command.
3. Stop Oracle High Availability Services on all the nodes of your cluster by executing the `crsctl stop crs` command on all the nodes as the `root` user.
4. Perform the restore by applying an OCR backup file that you identified in step 1 using the following command as the `root` user, where `file_name` is the name of the OCR file that you want to restore: `ocrconfig -restore file_name`.
Ensure that the OCR devices that you specify in the OCR configuration file (`/etc/oracle/ocr.loc`) exist and that these OCR devices are valid.
5. Restart Oracle High Availability Services on the nodes in your cluster by restarting each node or running the `crsctl start crs` command as the `root` user.
6. Check the integrity of the OCR files on all nodes with the `cluvfy comp ocr -n all` command.

Recovering the OCR by Using Logical Backups

1. Locate a logical backup created using an OCR export.
2. Stop Oracle Clusterware on all nodes:

```
# crsctl stop cluster -all
```

3. Stop Oracle High Availability Services on all nodes:

```
# crsctl stop crs
```

4. Restore the logical OCR backup:

```
# ocrconfig -import /shared/export/ocrback.dmp
```

5. Restart Oracle High Availability Services on all nodes:

```
# crsctl start crs
```

6. Check the OCR integrity: \$ cluvfy comp ocr -n all

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the following procedure to import the OCR on UNIX-based systems:

1. Identify the OCR export file that you want to import by identifying the OCR export file that you previously created using the `ocrconfig -export file_name` command.
2. Stop Oracle Clusterware on all the nodes in your RAC database by executing the `crsctl stop crs` command on the nodes as the `root` user.
3. Perform the import by applying an OCR export file that you identified in step 1 using the following command, where `file_name` is the name of the OCR file from which you want to import the OCR information:
`ocrconfig -import file_name`
4. Restart Oracle High Availability Services on all the nodes in your cluster by restarting each node by using the `crsctl start crs` command.
5. Run the following Cluster Verify Utility (`cluvfy`) command to verify the OCR integrity, where the `-n all` argument retrieves a listing of all the cluster nodes that are configured as part of your cluster:
`cluvfy comp ocr -n all`

Oracle Local Registry

- Each cluster node has a local registry for node-specific resources, called an Oracle Local Registry (OLR).
- The OLR is installed and configured when Oracle Clusterware is installed.
- One of its functions is to facilitate Clusterware startup in situations where the ASM stores the OCR and voting disks
- You can check the status of OLR by using `ocrcheck`:

```
$ ocrcheck -local
Status of Oracle Local Registry is as follows :
  Version          :      3
  Total space (kbytes)   : 262120
  Used space (kbytes)    :    2204
  Available space(kbytes): 259916
  ID                 : 1535380044
  Device/File Name     : /u01/app/11.2.0/grid/cdata/host01.olr
                        Device/File integrity check succeeded
                        Local registry integrity check succeeded
                        Logical corruption check succeeded
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In Oracle Clusterware 11g Release 2, each node in a cluster has a local registry for node-specific resources, called an Oracle Local Registry (OLR), that is installed and configured when Oracle Clusterware installs OCR. Multiple processes on each node have simultaneous read and write access to the OLR particular to the node on which they reside, regardless of whether Oracle Clusterware is running or fully functional.

The OLR provides various Oracle Clusterware processes with access to key configuration information even when Oracle Clusterware is not running on the node. One of its functions is to facilitate the Oracle Clusterware startup process in situations where the ASM stores the OCR and voting disks. During the startup process, the OLR is referenced to determine the exact location of the voting disks. This enables the node to join the cluster. After this initial phase, ASM is started. After ASM is started, processes that require the full OCR can start and the clusterware startup process completes.

By default, OLR is located at `grid_home/cdata/hostname.olr`. You can manage the OLR using `ocrcheck`, `ocrdump`, and `ocrconfig` utilities with the `-local` option.

You can check the status of OLR using the `ocrcheck` utility, as follows:

```
$ ocrcheck -local
```

You can display the content of OLR to the text terminal that initiated the program using the OCRDUMP utility, as follows:

```
$ ocrdump -local -stdout
```

You can perform administrative tasks on OLR using the OCRCONFIG utility. To export OLR to a file:

```
$ ocrconfig -local -export file_name
```

To import a specified file to OLR:

```
$ ocrconfig -local -import file_name
```

To modify the OLR file on the local node:

```
$ ocrconfig -local -repair olr file_name
```

The *olr* keyword used with the *-repair* option is valid only when *-local* is used.

Summary

In this lesson, you should have learned how to:

- Display Clusterware management proficiency
- Demonstrate OCR backup and recovery techniques



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only

Real Application Clusters Database Installation

16

ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to:

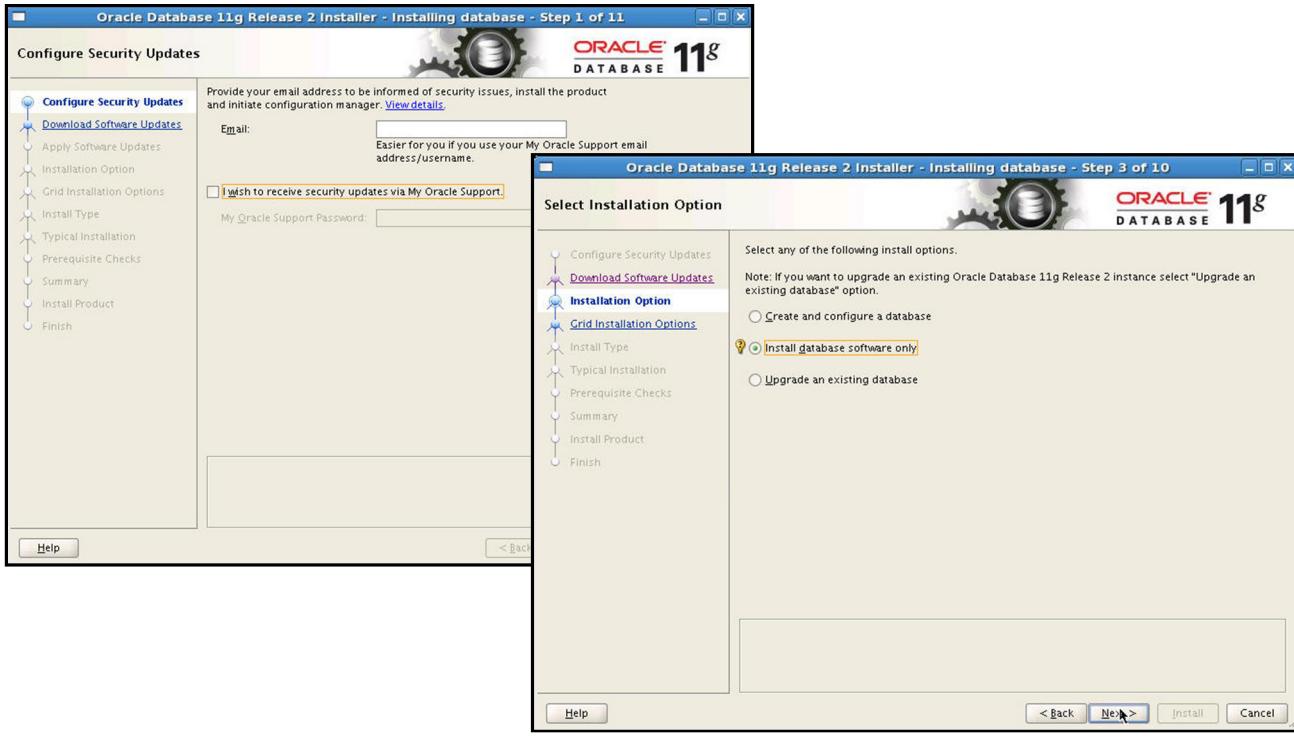
- Install the Oracle database software
- Create a cluster database
- Perform post-database-creation tasks



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Installing the Oracle Database Software

```
$ /stage/database/Disk1/runInstaller
```



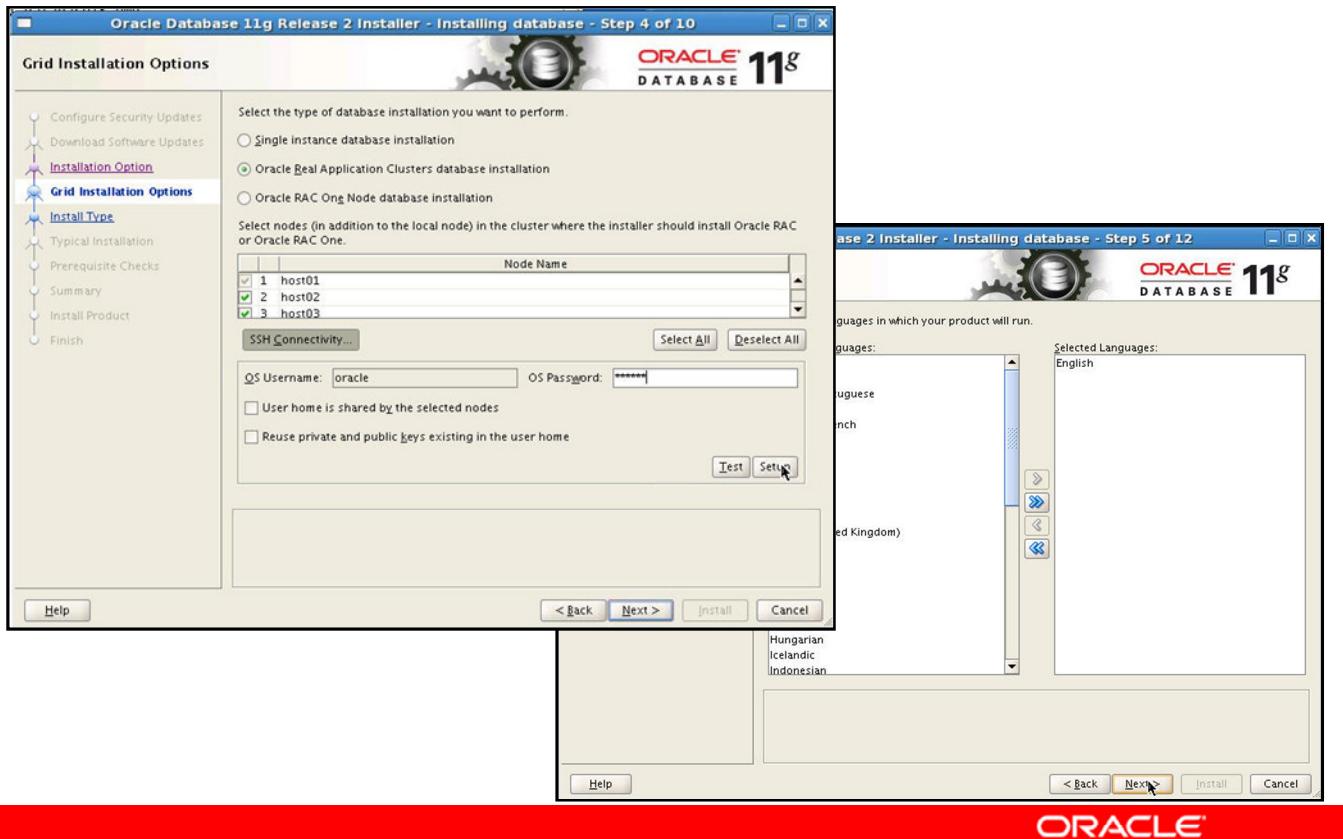
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Oracle Universal Installer (OUI) is used to install the Oracle Database 11g Release 2 (11.2) software. Start the OUI by executing the `runInstaller` command from the root directory of the Oracle Database 11g Release 2 CD-ROM or from the software staging location. You can use the Configure Security Updates window to specify an email address to receive security updates directly from Oracle Support as they occur. Alternatively, you can elect to opt out of these alerts. If you want to receive them, supply your email address and your Oracle Support password, and click Next.

The Download Software Updates page allows you to include database software updates or patches in the installation. The page allows you to either download them directly or use pre-downloaded updates. Alternatively, you can choose to bypass software updates entirely.

The Select Installation Option window enables you to create and configure a database, install database software only, or upgrade an existing database. Select the “Install database software only” option and click Next.

Installing the Oracle Database Software

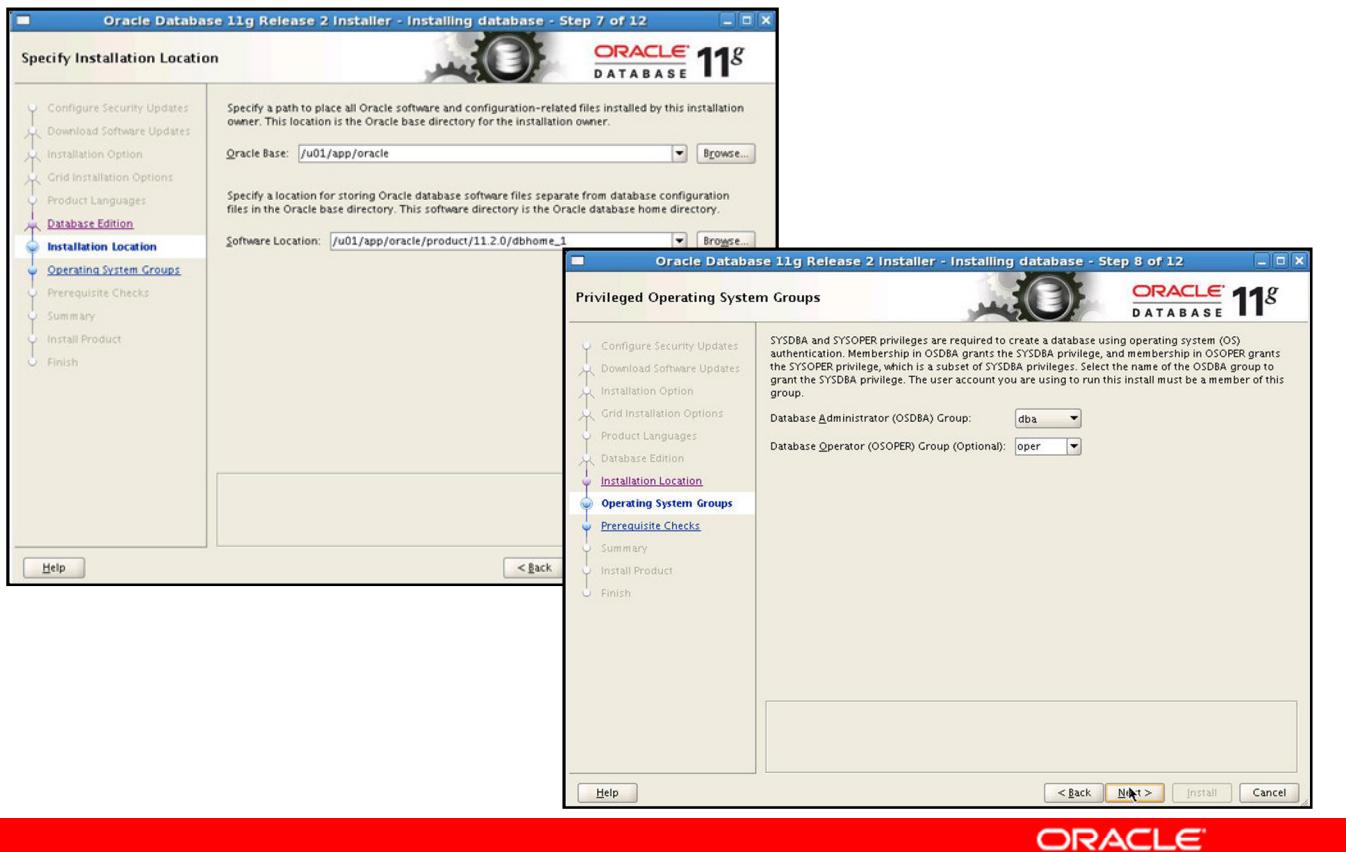


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the Grid Installation Options window, select “Real Application Clusters database installation” and select all nodes in your cluster on which the software should be installed. If SSH for the oracle user has not been set up, click SSH Connectivity, then provide the oracle user’s password and click Setup. If SSH has already been set up, then click Test. When finished, click Next to continue. In the “Select product languages” window, select your desired languages from the Available Languages list and click the right arrow to promote the selected languages to the Selected Languages list. Click Next to continue.

In the “Select database edition” window (not shown in the slide), you select whether to install the Enterprise Edition or the Standard Edition. Select the Enterprise Edition option and click Next to continue.

Installing the Oracle Database Software



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On the Select Database edition page, you can select to install either the Enterprise Edition or Standard Edition. Click Next to continue.

In the Specify Installation Location window, provide a value for ORACLE_BASE if you have not already done so. The default ORACLE_BASE location is /u01/app/oracle, provided the RDBMS software is being installed by the oracle account. The Software Location section of the window enables you to specify a value for the ORACLE_HOME location. The default ORACLE_HOME location is /u01/app/oracle/product/11.2.0/dbhome_1. Accept the suggested path or enter your own location. After entering the information, review it for accuracy, and click the Next button to continue. In the "Privileged operating system groups" window, select the operating system group that will act as the OSDBA group. Next, select the group that will act as the OSOPER group. Click Next to continue.

Installing the Oracle Database Software



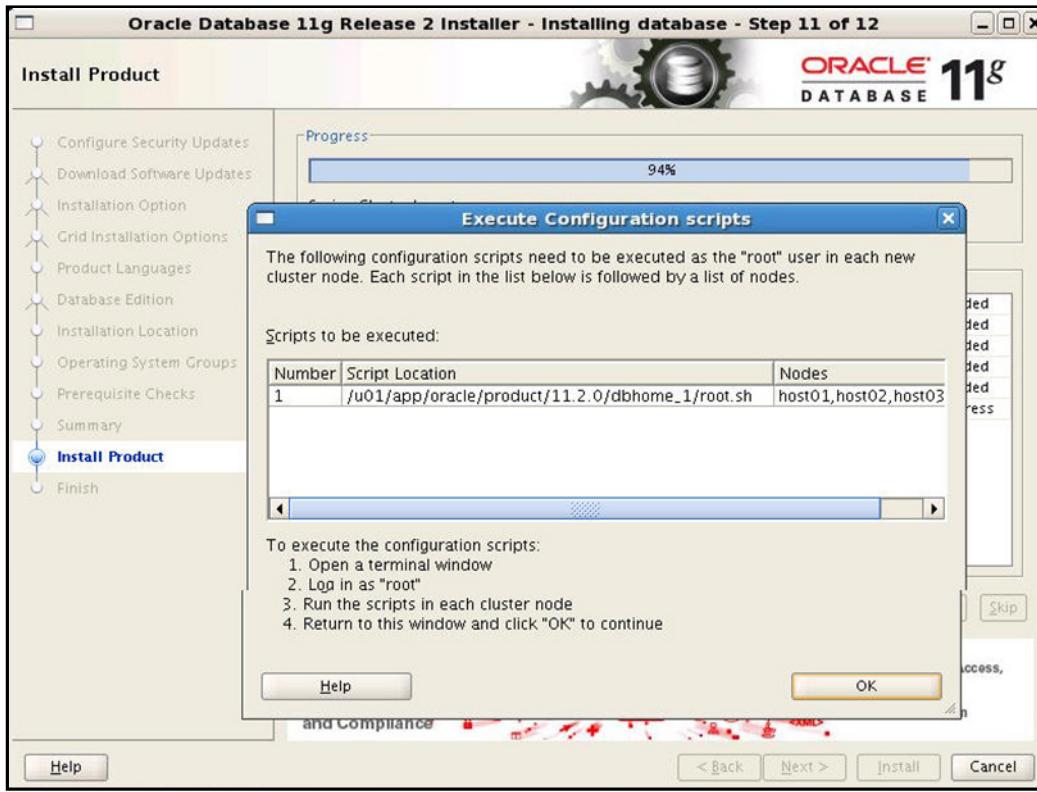
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The “Perform Prerequisite Checks” window verifies the operating system requirements that must be met for the installation to be successful. These requirements include:

- Certified operating system check
- Kernel parameters as required by the Oracle Database software
- Required operating system packages and correct revisions

After each successful check, the Status for that check will indicate Succeeded. Any tests that fail are also reported here. If any tests fail, click the “Fix & Check Again” button. The Installer will generate fix-up scripts to correct the system deficiencies if possible. Execute the scripts as directed by the Installer. The tests will be run again after completing the script executions. When all tests have succeeded, click the Next button to continue. In the Summary window, review the Global settings and click Finish.

Installing the Oracle Database Software

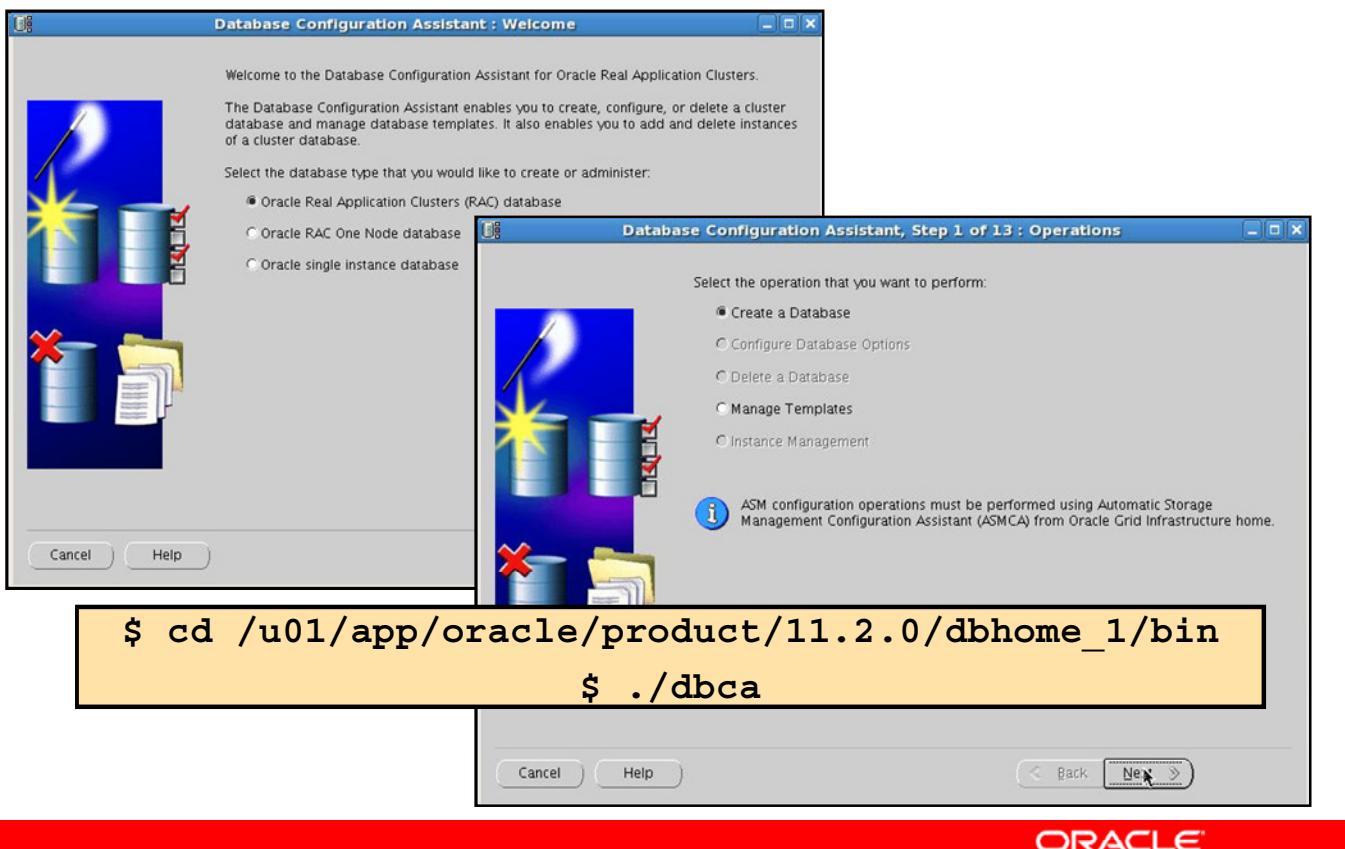


ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

At the end of the installation, the OUI will display another window, prompting you to run the `root.sh` scripts on the nodes you chose for the installation. Follow the instructions to run the scripts. When finished, click the OK button to close the Execute Configuration Scripts window and return to the Finish screen. Click Close to complete the installation and close the OUI.

Creating the Cluster Database



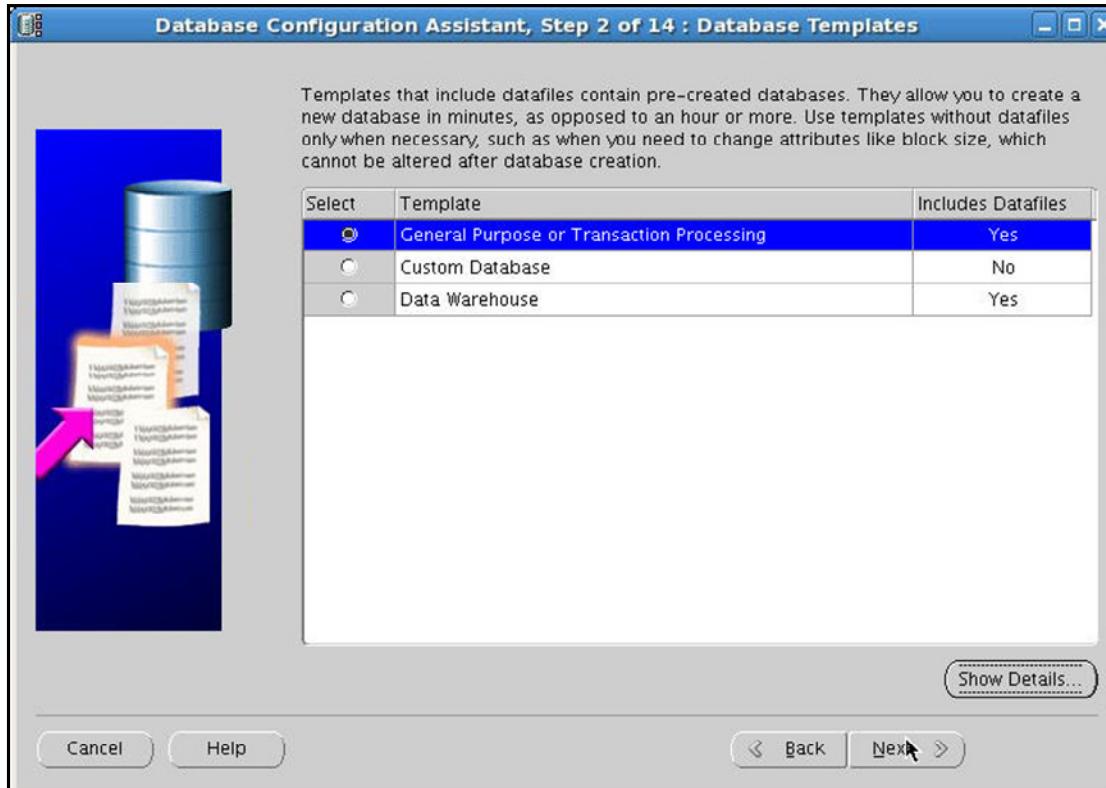
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To create the cluster database, change directory to \$ORACLE_HOME/bin on the installing node and execute the database configuration assistant (DBCA) utility as follows:

```
$ cd /u01/app/oracle/product/11.2.0/dbhome_1/bin  
$ ./dbca
```

The Welcome window appears first. You must select the type of database that you want to install. Select the “Oracle Real Application Clusters (RAC) database” option, and then click Next. The Operations window appears. For a first-time installation, you have only two choices: the first option enables you to create a database and the other option enables you to manage database creation templates. Select the “Create a Database” option, and then click Next to continue.

Database Type Selection



ORACLE

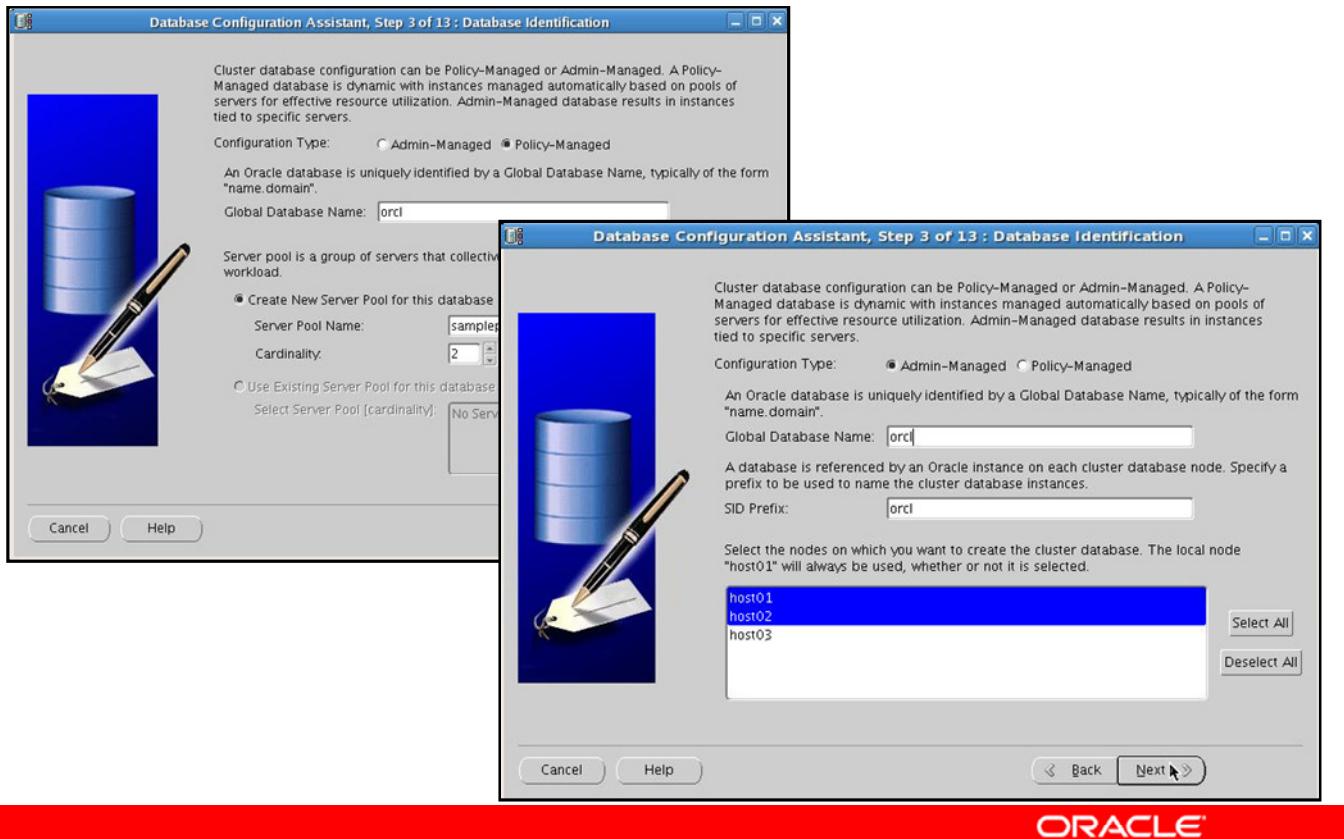
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Database Templates window appears next. The DBCA tool provides several predefined database types to choose from, depending on your needs. The templates include:

- General Purpose or Transaction Processing
- Custom Database
- Data Warehouse

In the example in the slide, the “General Purpose or Transaction Processing” option is chosen. Click Next to continue.

Database Identification



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

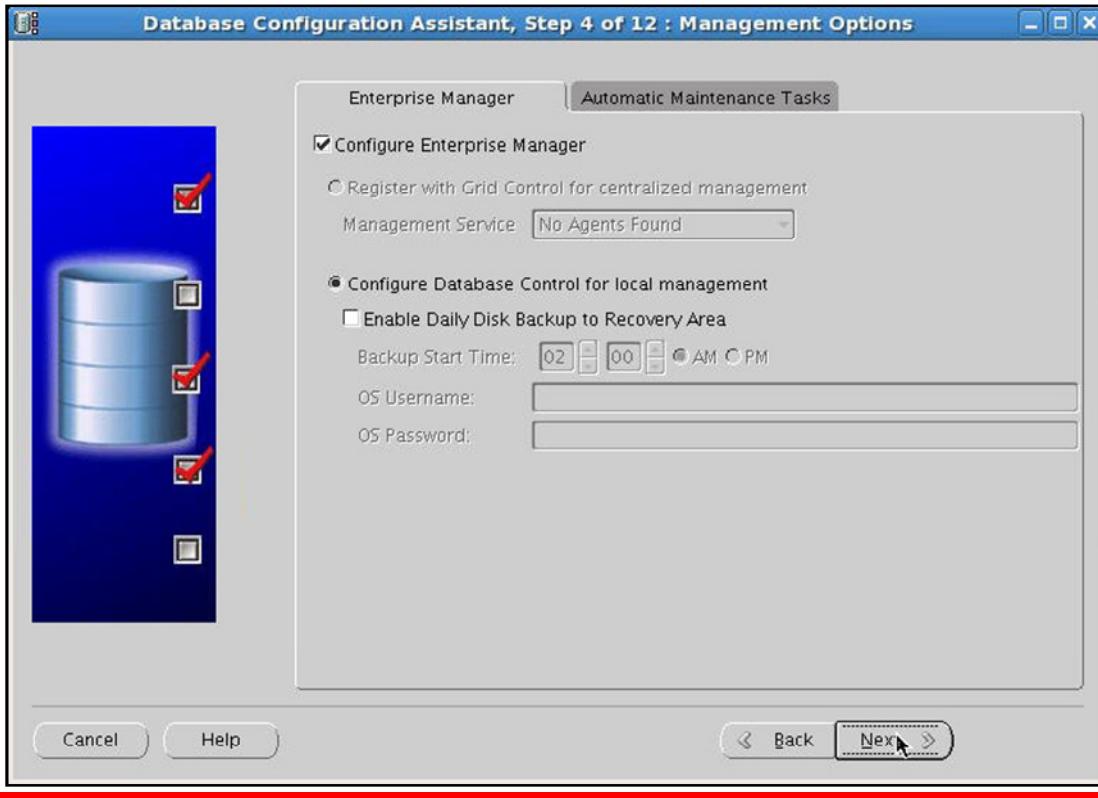
In the Database Identification window, you must choose between an administrator-managed and a policy-managed cluster database.

Administrator-managed RAC databases specify a list of cluster nodes where RAC instances will run. Services may also be specified and associated with preferred and alternative nodes. There is an explicit association between database services, instances, and cluster nodes.

Policy-based management, a new feature in this release, breaks the explicit association between services, instances, and cluster nodes. Policy-based management introduces the concept of server pools, which are logical divisions of a cluster that are dynamically allocated based on relative importance. Database services are associated with server pools, and RAC instances are automatically started to satisfy the service to server pool associations. You specify in which server pool the database resource will run and the number of instances needed (cardinality). Oracle Clusterware is responsible for placing the database resource on a server. Server pools are logical divisions of a cluster into pools of servers that are allocated to host databases or other applications. Server pools are managed using `crsctl` and `srvctl` commands. Names must be unique within the resources defined for the cluster.

You must also choose the global database name and the nodes on which to create the cluster database. The global database name can be up to 30 characters in length and must begin with an alphabetical character. When you have finished, click Next to continue.

Cluster Database Management Options



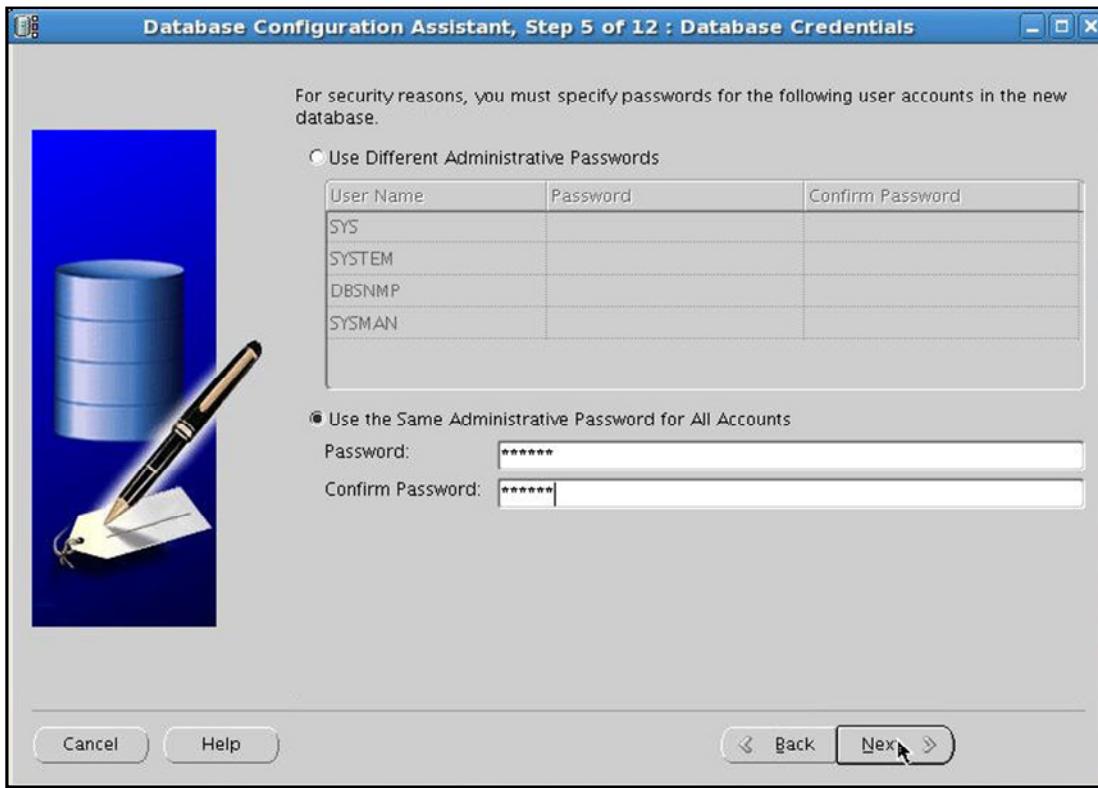
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Management Options window is displayed. For small cluster environments, you may choose to manage your cluster with Enterprise Manager Database Control. To do this, select the “Configure Enterprise Manager” check box. If you have Grid Control installed somewhere on your network, you can select the “Use Grid Control for Database Management” option. If you select Enterprise Manager with the Grid Control option and the DBCA discovers agents running on the local node, you can select the preferred agent from a list. Grid Control can simplify database management in large, enterprise deployments.

You can also configure Database Control to send email notifications when alerts occur. If you want to configure this, you must supply a Simple Mail Transfer Protocol (SMTP) or outgoing mail server and an email address. You can also enable daily backups here. You must supply a backup start time as well as operating system user credentials for this option.

If you want to use Grid Control to manage your database, but have not yet installed and configured a Grid Control server, do not click either of the management methods. When you have made your choices, click the Next button to continue.

Passwords for Database Schema Owners



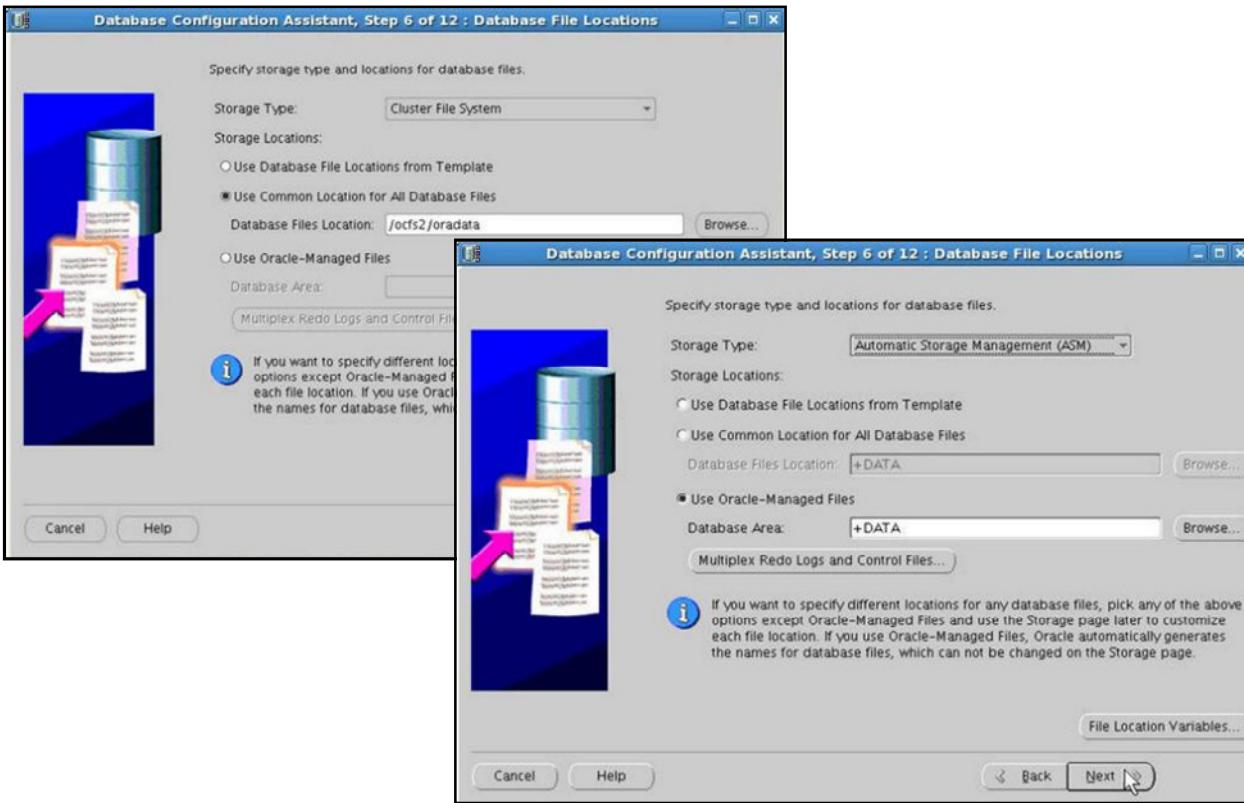
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ORACLE

The Database Credentials window appears next. You must supply passwords for the user accounts created by the DBCA when configuring your database. You can use the same password for all of these privileged accounts by selecting the “Use the Same Administrative Password for All Accounts” option. Enter your password in the Password field, and then enter it again in the Confirm Password field.

Alternatively, you may choose to set different passwords for the privileged users. To do this, select the “Use Different Administrative Passwords” option, enter your password in the Password field, and then enter it again in the Confirm Password field. Repeat this for each user listed in the User Name column. Click the Next button to continue.

Database File Locations

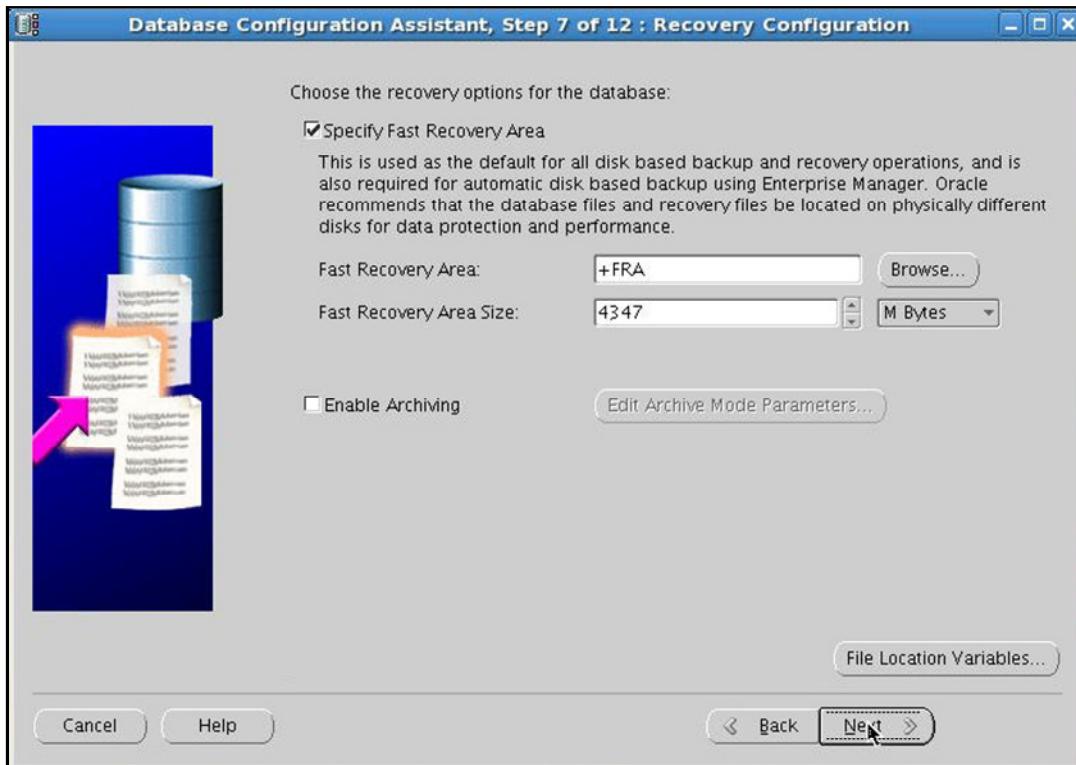


ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the Database File Locations window, you must indicate where the database files are to be stored. You can choose your storage type from the drop-down list. Detected (and supported) shared storage types are available here. You can choose to use a standard template for file locations, one common location, or Oracle-Managed Files (OMF). This cluster database uses ASM and Oracle Managed Files. Therefore, select the Use Oracle-Managed Files option, and enter the disk group name in the Database Area field. Alternatively, you can click the Browse button to indicate the location where the database files are to be created. When you have made your choices, click the Next button to continue.

Recovery Configuration



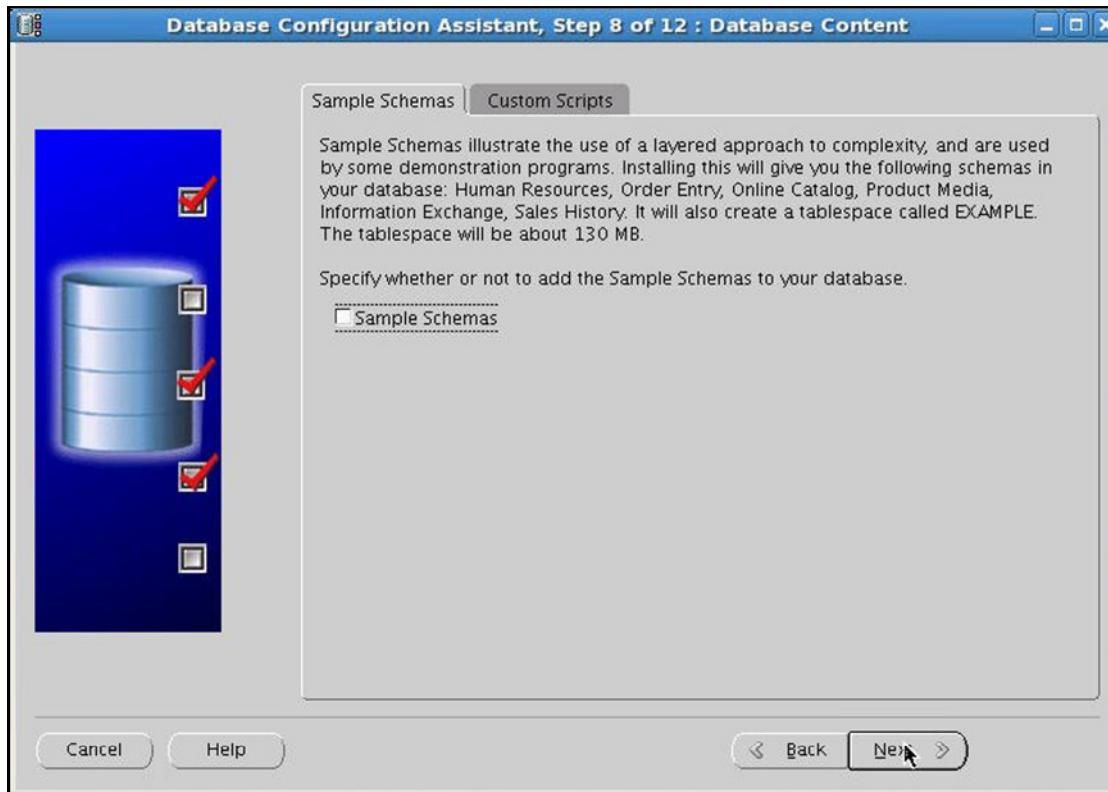
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the Recovery Configuration window, you can select redo log archiving by selecting Enable Archiving. If you are using ASM or cluster file system storages, you can also select the fast recovery area size in the Recovery Configuration window. The size of the area defaults to 2048 megabytes, but you can change this figure if it is not suitable for your requirements. If you are using ASM and a single disk group, the fast recovery area defaults to the ASM Disk Group. If more than one disk group has been created, you can specify it here. If you use a cluster file system, the fast recovery area defaults to \$ORACLE_BASE/flash_recovery_area. You may also define your own variables for the file locations if you plan to use the Database Storage window to define individual file locations. Select the Enable Archiving check box to enable archiving immediately for the new cluster database.

When you have completed your entries, click Next, and the Database Content window is displayed.

Note: For Oracle Database 11g Release 2 (11.2), the flash recovery area has been renamed “fast recovery area.” Oracle Enterprise Manager, however, still uses the older vocabulary on its webpages.

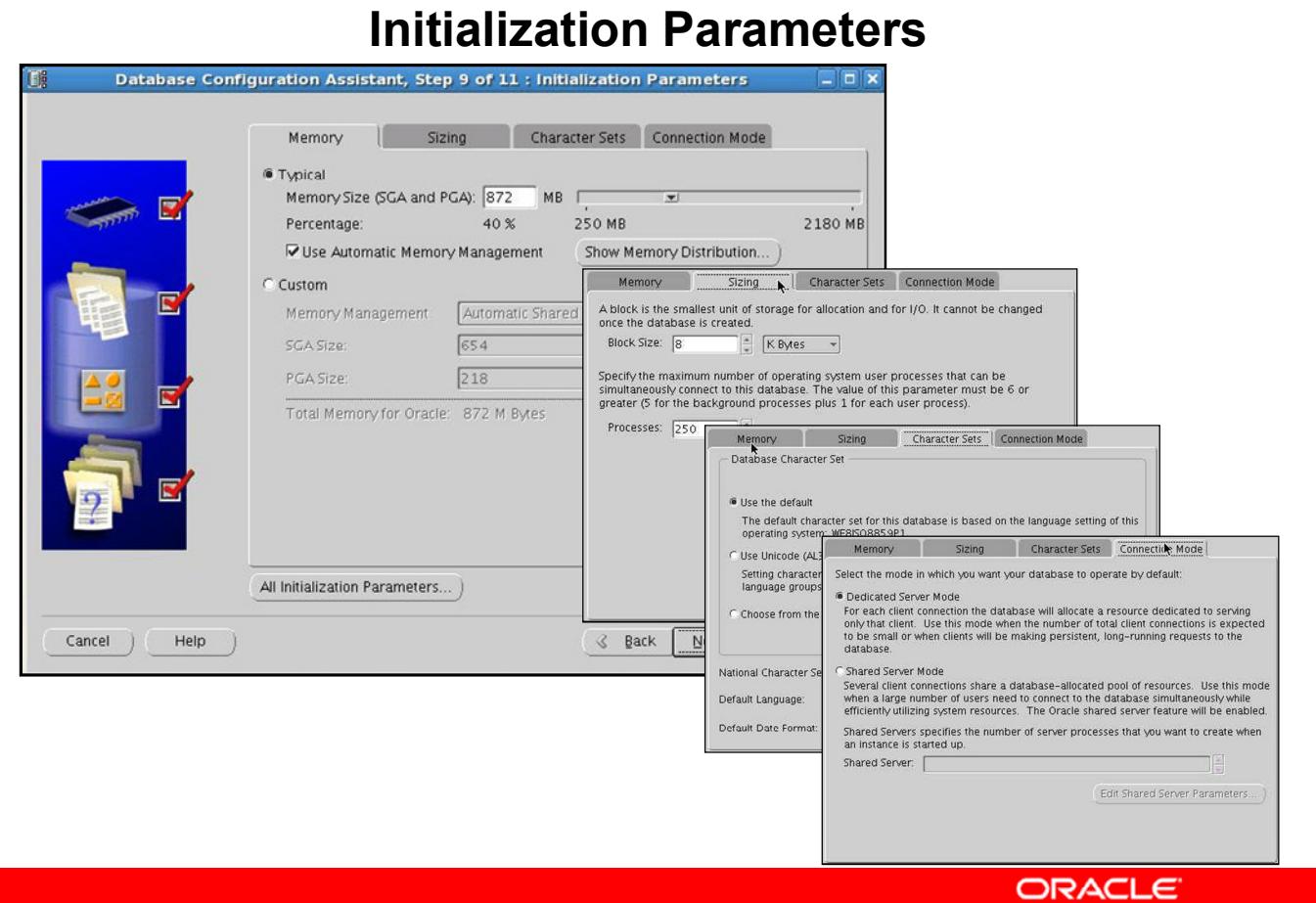
Database Content



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the Database Content window, you can choose to install the Sample Schemas included with the database distribution. On the Custom Scripts tabbed page, you can choose to run your own scripts as part of the database creation process. When you have finished, click the Next button to continue to the next window.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

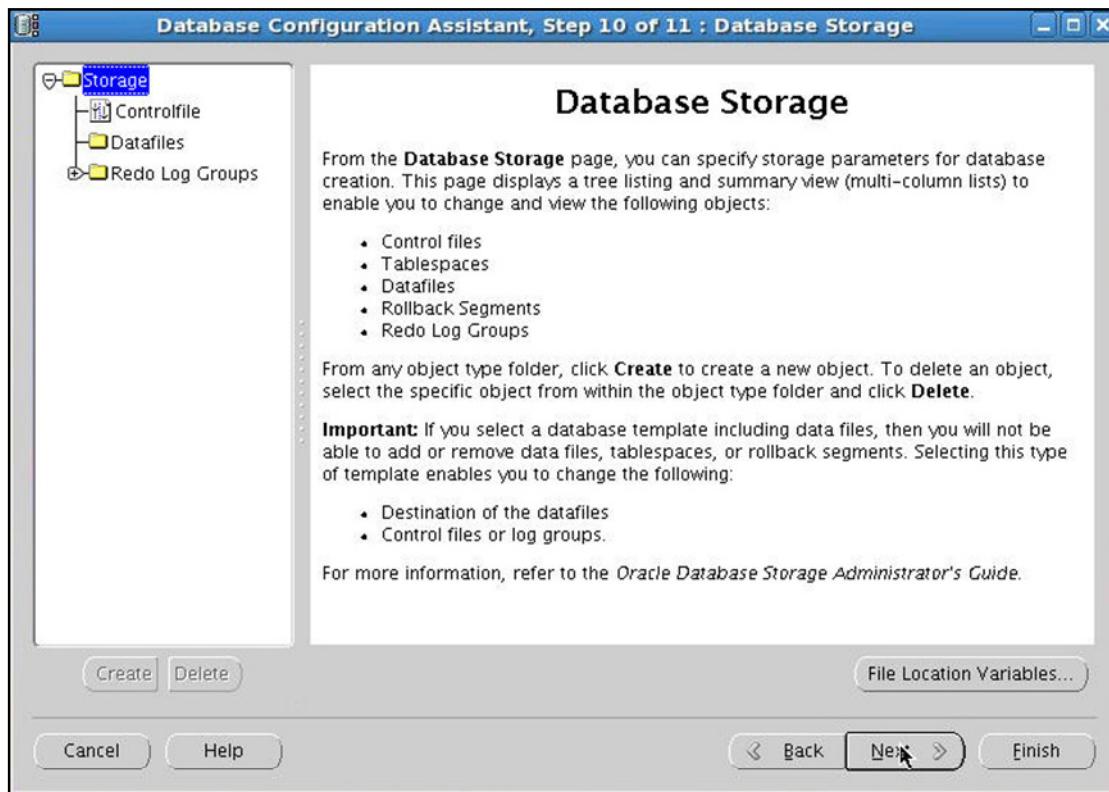
In the Initialization Parameters window, you can set important database parameters. The parameters are grouped under four tabs:

- Memory
- Sizing
- Character Sets
- Connection Mode

On the Memory tabbed page, you can set parameters that deal with memory allocation, including shared pool, buffer cache, Java pool, large pool, and PGA size. Automatic Memory Management is the preferred memory management method and can be selected here. On the Sizing tabbed page, you can adjust the database block size. Note that the default is 8 KB. In addition, you can set the number of processes that can connect simultaneously to the database.

By clicking the Character Sets tab, you can change the database character set. You can also select the default language and the date format. On the Connection Mode tabbed page, you can choose the connection type that clients use to connect to the database. The default type is Dedicated Server Mode. If you want to use Oracle Shared Server, click the Shared Server Mode button. If you want to review the parameters that are not found on the four tabs, click the All Initialization Parameters button. Click the Use Automatic Memory Management button and click the Next button to continue.

Database Storage Options

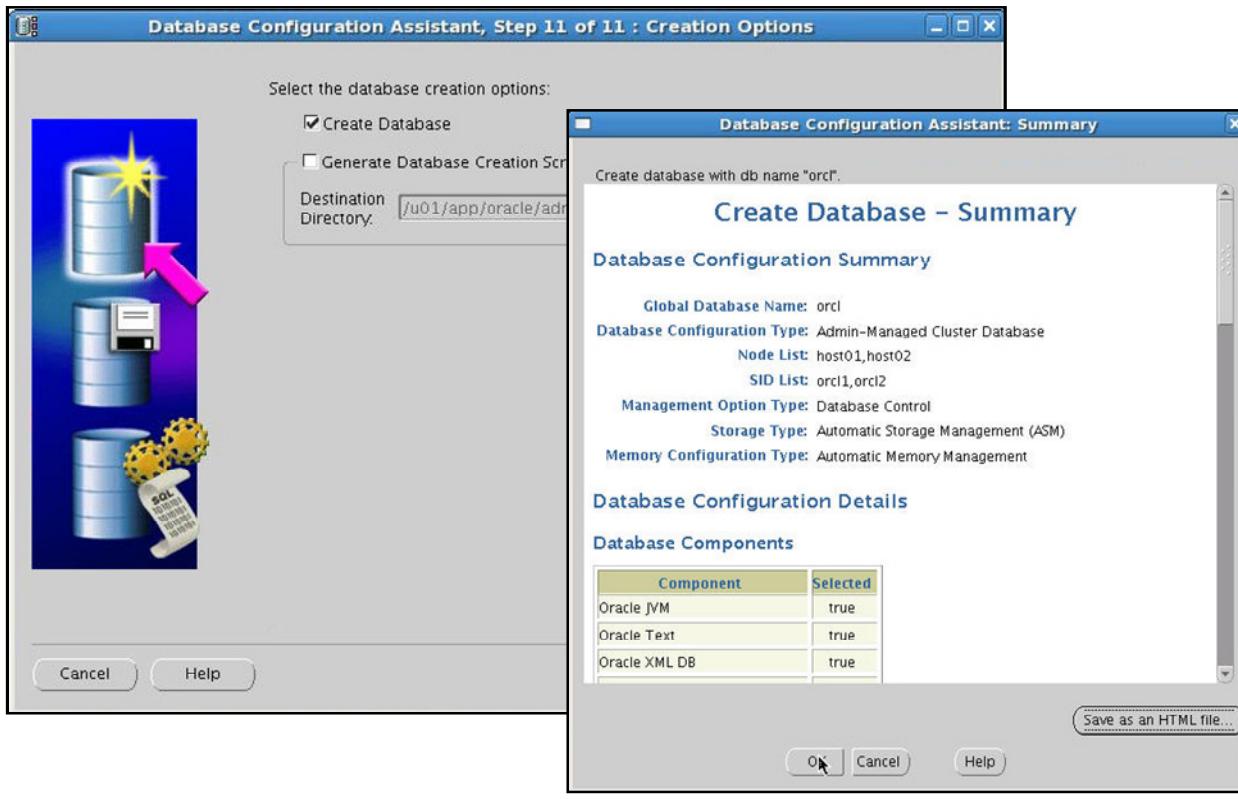


Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Database Storage window provides full control over all aspects of database storage, including tablespaces, data files, and log members. Size, location, and all aspects of extent management are under your control here.

When you have finished, click the Next button to continue to the next page.

Create the Database



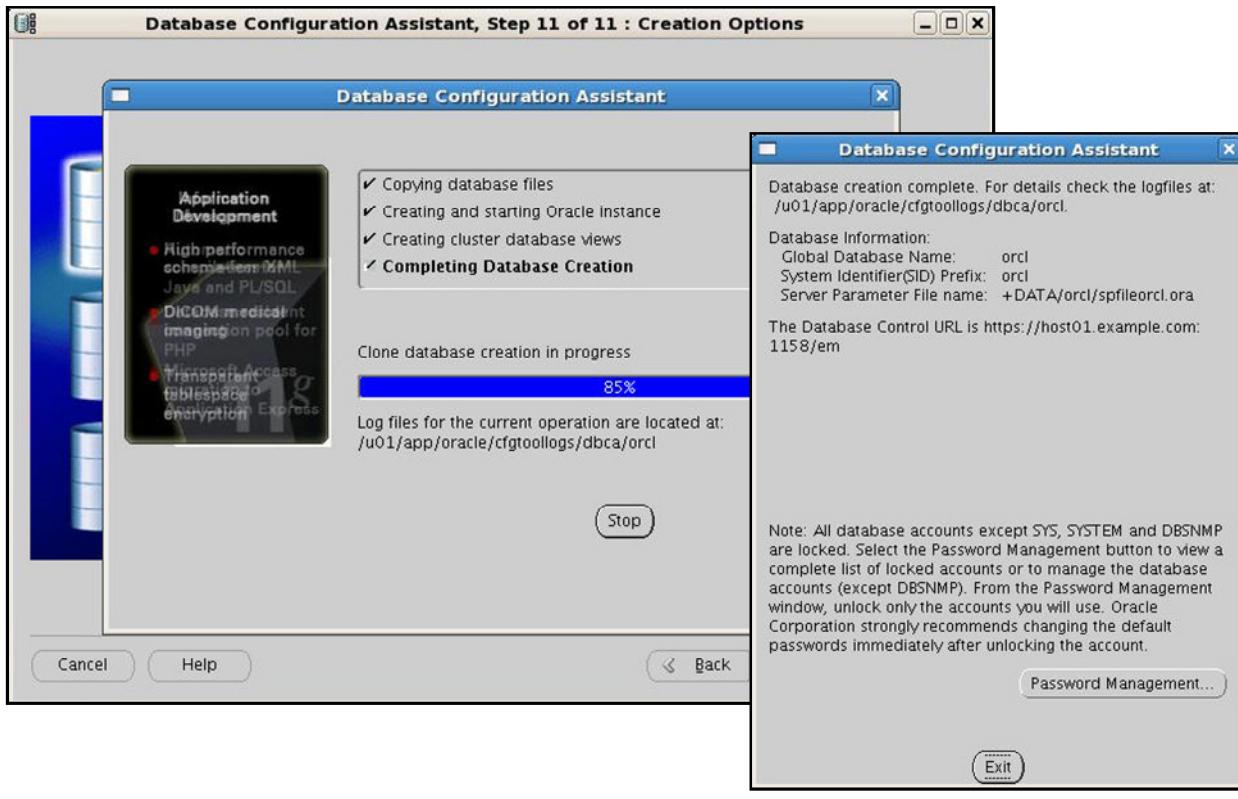
ORACLE®

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Creation Options window appears. You can choose to create the database, or save your DBCA session as a database creation script by selecting the corresponding check box. Select the Create Database check box, and then click the Finish button. The DBCA displays the Summary screen, giving you a last chance to review all options, parameters, and so on that have been chosen for your database creation.

Review the summary data. When you are ready to proceed, close the Summary window by clicking the OK button.

Monitoring Progress



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Progress Monitor window appears next. In addition to informing you about how fast the database creation is taking place, it also informs you about the specific tasks being performed by the DBCA in real time. When the database creation progress reaches 100 percent, the DBCA displays a dialog box announcing the completion of the creation process. It also directs you to the installation log file location, parameter file location, and Enterprise Manager URL. By clicking the Password Management button, you can manage the database accounts created by the DBCA.

Postinstallation Tasks

- Download and install the required patch updates.
- Verify the cluster database configuration.

```
$ 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: PRIMARY
Management policy: AUTOMATIC
Server pools: orcl
Database instances: orcl1,orcl2
Disk Groups: DATA,FRA
Services:
Type: RAC
Database is administrator managed
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

After the cluster database has been successfully created, run the following command to verify the Oracle Cluster Registry configuration in your newly installed RAC environment:

```
$ srvctl config database -d db_name
```

Summary

In this lesson, you should have learned how to:

- Install the Oracle database software
- Create a cluster database
- Perform post-database-creation tasks



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

THESE eKIT MATERIALS ARE FOR YOUR USE IN THIS CLASSROOM ONLY. COPYING eKIT MATERIALS FROM THIS COMPUTER IS STRICTLY PROHIBITED

Oracle University and Error : You are not a Valid Partner use only