Oracle Database 11*g*: Administration Workshop I

Volume 2 - Student Guide

D50102GC10

Edition 1.0

September 2007

D52684 L$

Authors

Priya Vennapusa
James Spiller
Maria Billings

Technical Contributors and Reviewers

Christian Bauwens
Sangram Dash
Andy Fortunak
Gerlinde Frenzen
Steve Friedberg
Joel Goodman
Magnus Isaksson
Akira Kinutani
Pete Jones
Pierre Labrousse
Gwen Lazenby
Hakan Lindfors
Srinivas Putrevu
Andreas Reinhardt
Ira Singer
Jenny Tsai

Editors

Richard Wallis
Amitha Narayan

Graphic Designer

Rajiv Chandrabhanu

Publishers

Nita Brozowski
Michael Sebastian Almeida

Disclaimer

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

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

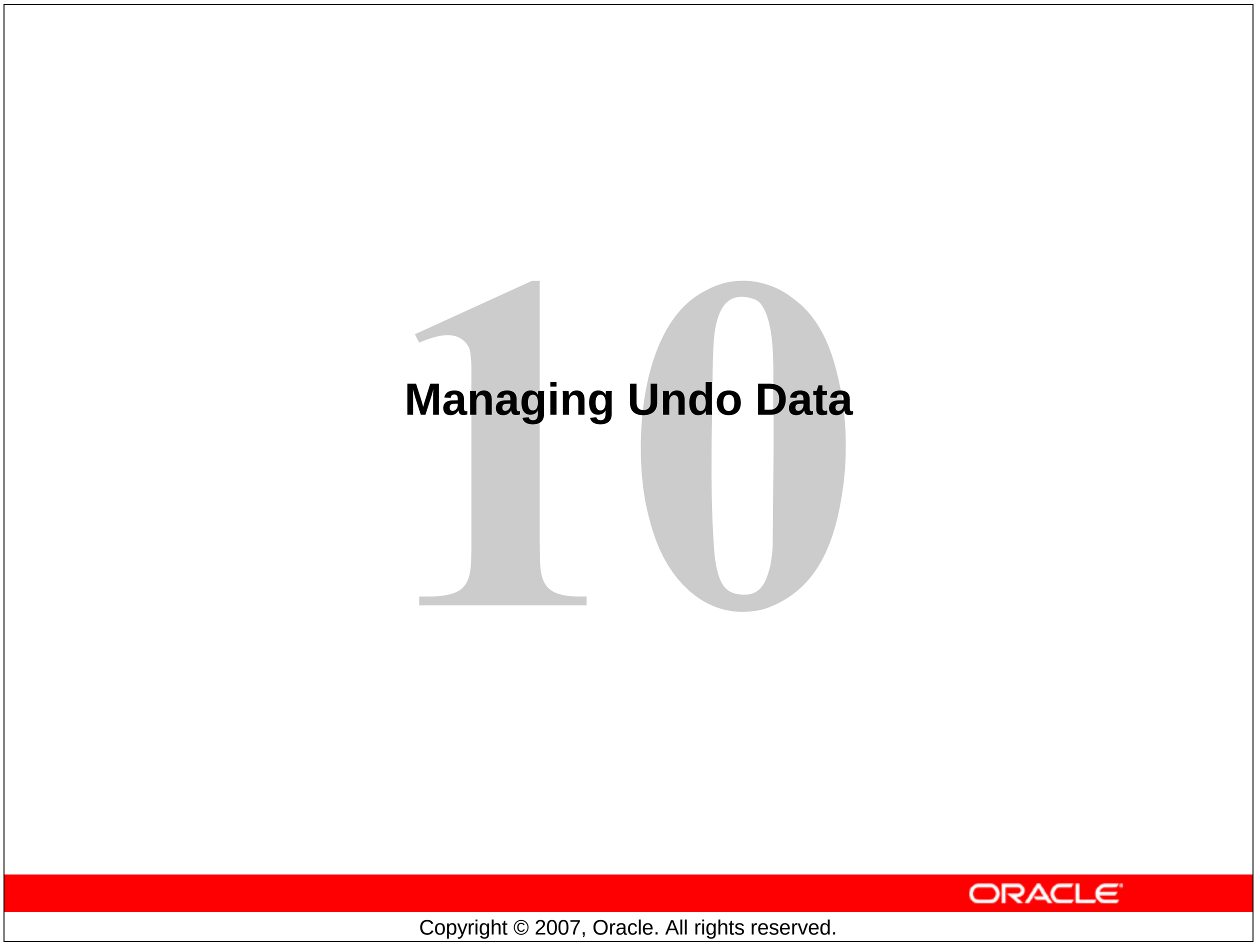
Restricted Rights Notice

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

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

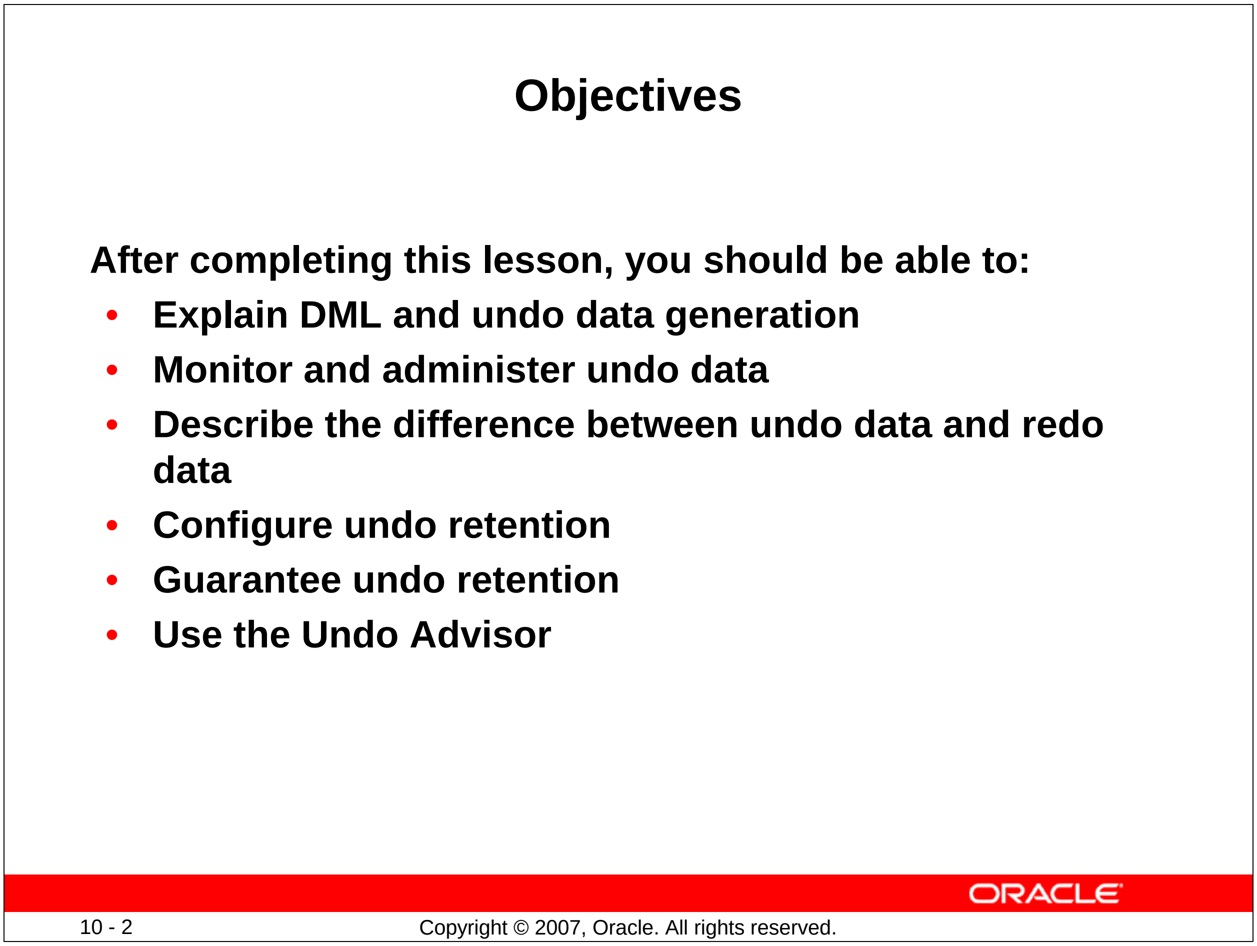
Trademark Notice

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



10

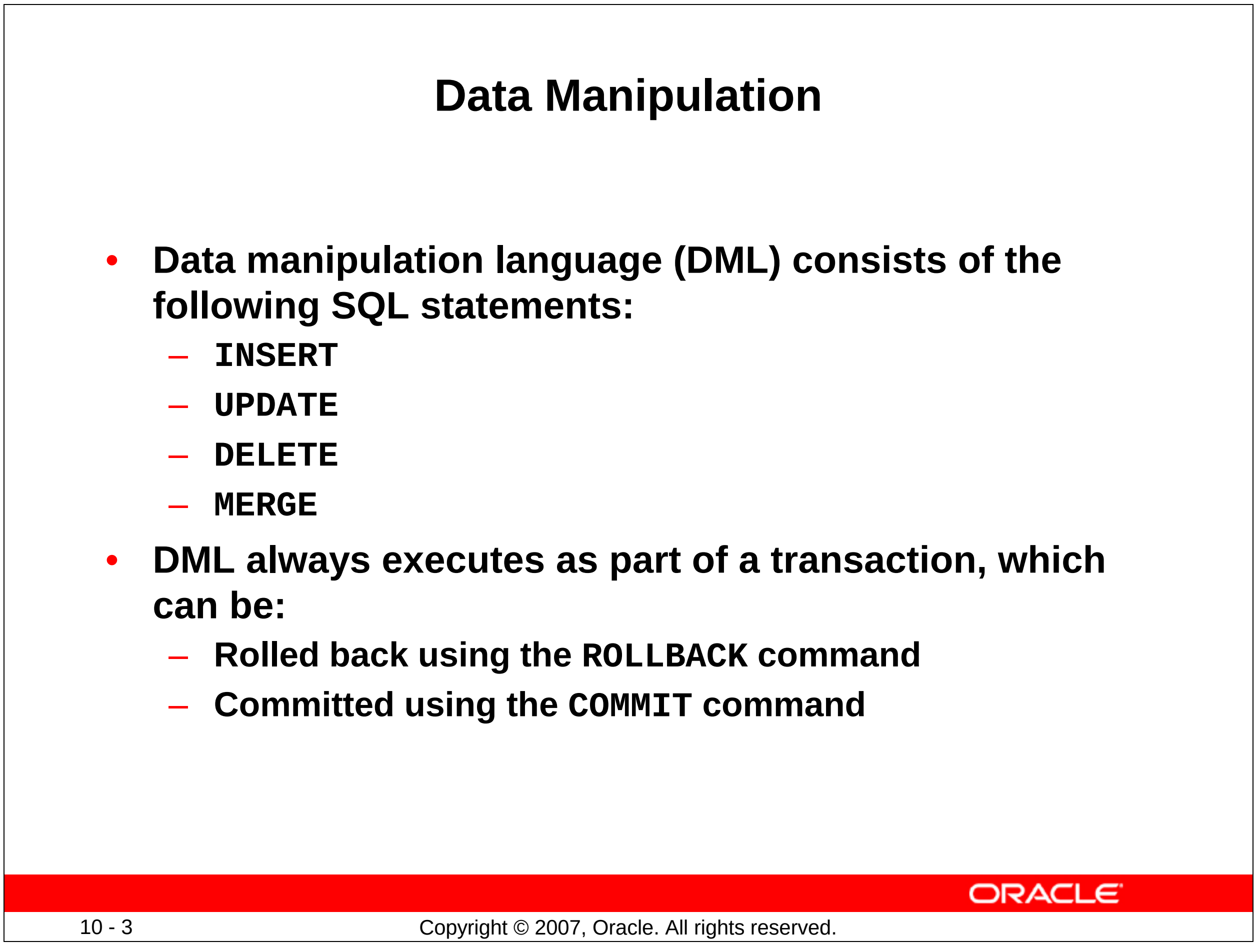
Managing Undo Data



Objectives

After completing this lesson, you should be able to:

- Explain DML and undo data generation
- Monitor and administer undo data
- Describe the difference between undo data and redo data
- Configure undo retention
- Guarantee undo retention
- Use the Undo Advisor



Data Manipulation

- Data manipulation language (DML) consists of the following SQL statements:
  - `INSERT`
  - `UPDATE`
  - `DELETE`
  - `MERGE`
- DML always executes as part of a transaction, which can be:
  - Rolled back using the `ROLLBACK` command
  - Committed using the `COMMIT` command

10 - 3

Data Manipulation

Data is manipulated, or modified, by the DML class of SQL statements: `INSERT`, `UPDATE`, `DELETE`, and `MERGE`. These statements execute as part of a transaction, which starts with the first successful DML statement and ends with either a `COMMIT` or `ROLLBACK` command. A transaction is either entirely committed or entirely rolled back.

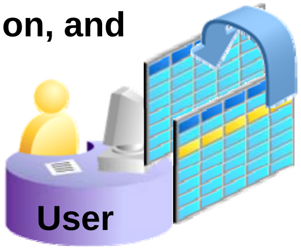
In addition to the explicit `COMMIT` or `ROLLBACK` commands, they can also occur implicitly. For example, rollback may occur if there is a process or system failure. Commit may occur after a DDL command, such as the `CREATE TABLE` command.

Note: The `MERGE` command performs a combination of DML commands to merge data from one table into another. It is covered in the lesson titled “Managing Data and Concurrency.”

Undo Data

Undo data is:

- **A copy of original, premodified data**
- **Captured for *every* transaction that changes data**
- **Retained at least until the transaction is ended**
- **Used to support:**
  - **Rollback operations**
  - **Read-consistent queries**
  - **Flashback Query, Flashback Transaction, and Flashback Table**
  - **Recovery from failed transactions**



Undo Data

The Oracle database saves the old value (undo data) when a process changes data in a database. It stores the data as it exists before modifications. Capturing undo data enables you to roll back your uncommitted data. Undo supports read-consistent and flashback queries. Undo can also be used to “rewind” (flash back) transactions and tables.

Read-consistent queries provide results that are consistent with the data as of the time a query started. For a read-consistent query to succeed, the original information must still exist as undo information. If the original data is no longer available, you receive a “Snapshot too old” error. As long as the undo information is retained, the Oracle database can reconstruct data to satisfy read-consistent queries.

Flashback queries purposely ask for a version of the data as it existed at some time in the past. As long as undo information for that past time still exists, flashback queries can complete successfully. Flashback Transaction uses undo to create compensating transactions, to back out a transaction and its dependent transactions. With Flashback Table, you can recover a table to a specific point in time.

Undo data is also used to recover from failed transactions. A failed transaction occurs when a user session ends abnormally (possibly because of network errors or a failure on the client computer) before the user decides to commit or roll back the transaction. Failed transactions may also occur when the instance crashes or you issue the `SHUTDOWN ABORT` command.

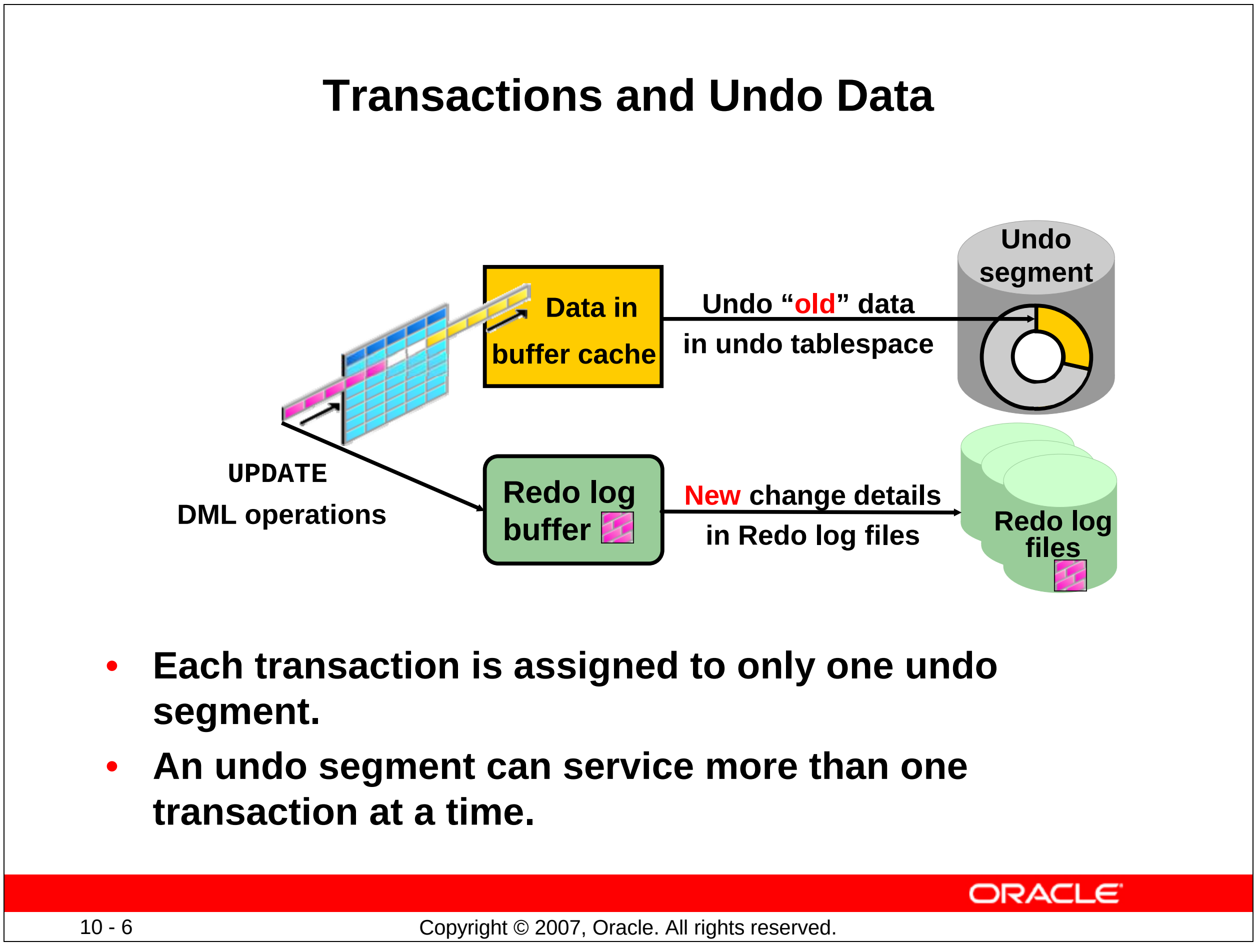
Undo Data (continued)

In case of a failed transaction, the safest behavior is chosen, and the Oracle database reverses all changes made by a user, thereby restoring the original data.

Undo information is retained for all transactions, at least until the transaction is ended by one of the following:

- User undoes a transaction (transaction rolls back).
- User ends a transaction (transaction commits).
- User executes a DDL statement, such as a `CREATE`, `DROP`, `RENAME` or `ALTER` statement. If the current transaction contains any DML statements, the database first commits the transaction and then executes and commits the DDL as a new transaction.
- User session terminates abnormally (transaction rolls back).
- User session terminates normally with an exit (transaction commits).

The amount of undo data that is retained and the time for which it is retained depend on the amount of database activity and the database configuration.



Transactions and Undo Data

- Each transaction is assigned to only one undo segment.
- An undo segment can service more than one transaction at a time.

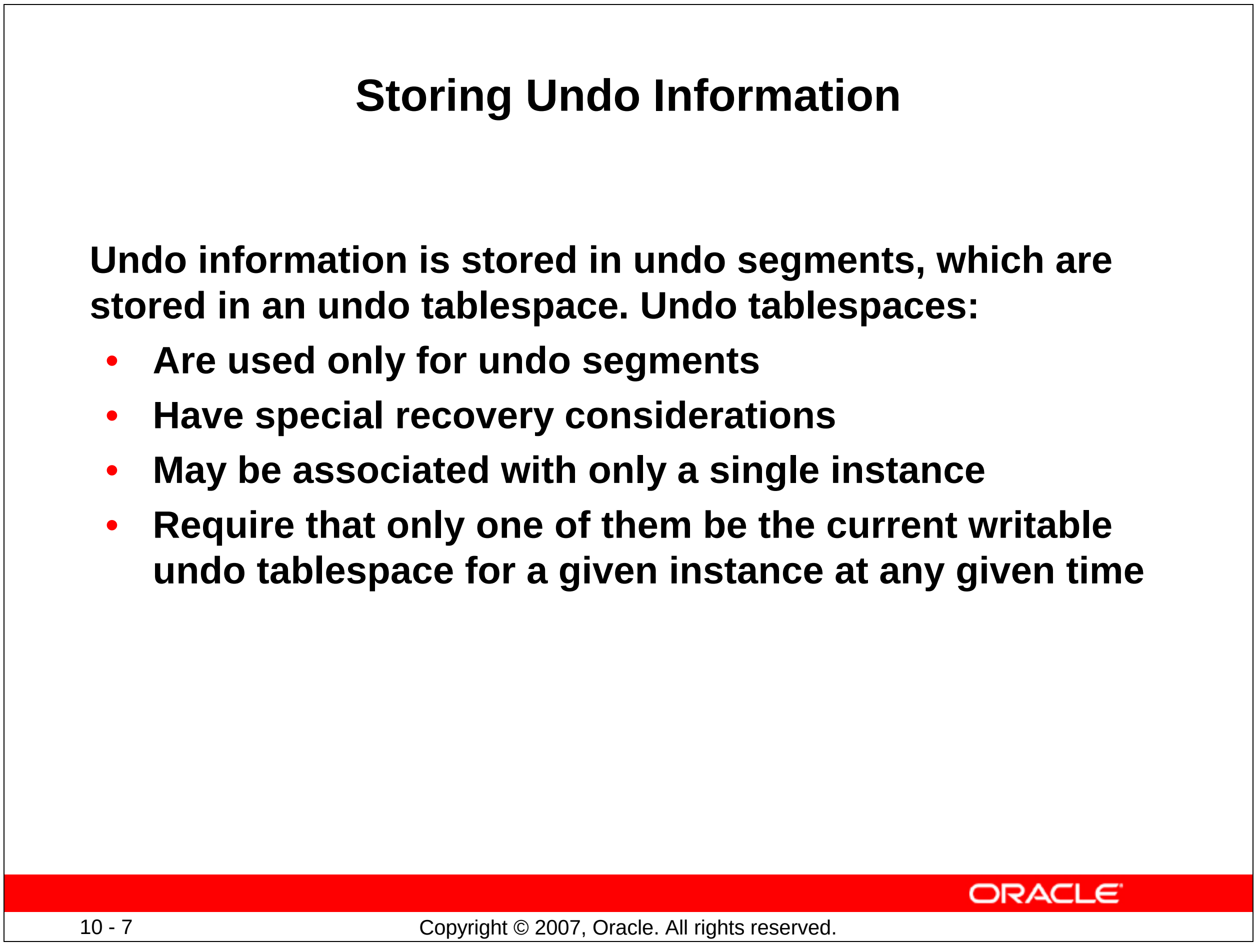
Transactions and Undo Data

When a transaction starts, it is assigned to an undo segment. Throughout the life of the transaction, when data is changed, the original (before the change) values are copied into the undo segment. You can see which transactions are assigned to which undo segments by checking the `V$TRANSACTION` dynamic performance view.

Undo segments are specialized segments that are automatically created by the instance as needed to support transactions. Like all segments, undo segments are made up of extents, which, in turn, consist of data blocks. Undo segments automatically grow and shrink as needed, acting as a circular storage buffer for their assigned transactions.

Transactions fill extents in their undo segments until a transaction is completed or all space is consumed. If an extent fills up and more space is needed, the transaction acquires that space from the next extent in the segment. After all extents have been consumed, the transaction either wraps around back into the first extent or requests a new extent to be allocated to the undo segment.

Note: Parallel DML and DDL operations can actually cause a transaction to use more than one undo segment. To learn more about parallel DML execution, see the *Oracle Database Administrator's Guide*.



Storing Undo Information

Undo information is stored in undo segments, which are stored in an undo tablespace. Undo tablespaces:

- **Are used only for undo segments**
- **Have special recovery considerations**
- **May be associated with only a single instance**
- **Require that only one of them be the current writable undo tablespace for a given instance at any given time**

10 - 7

Storing Undo Information

Undo segments can exist only in a specialized form of tablespace called an *undo tablespace*. (You cannot create other segment types, such as tables, in the undo tablespace.)

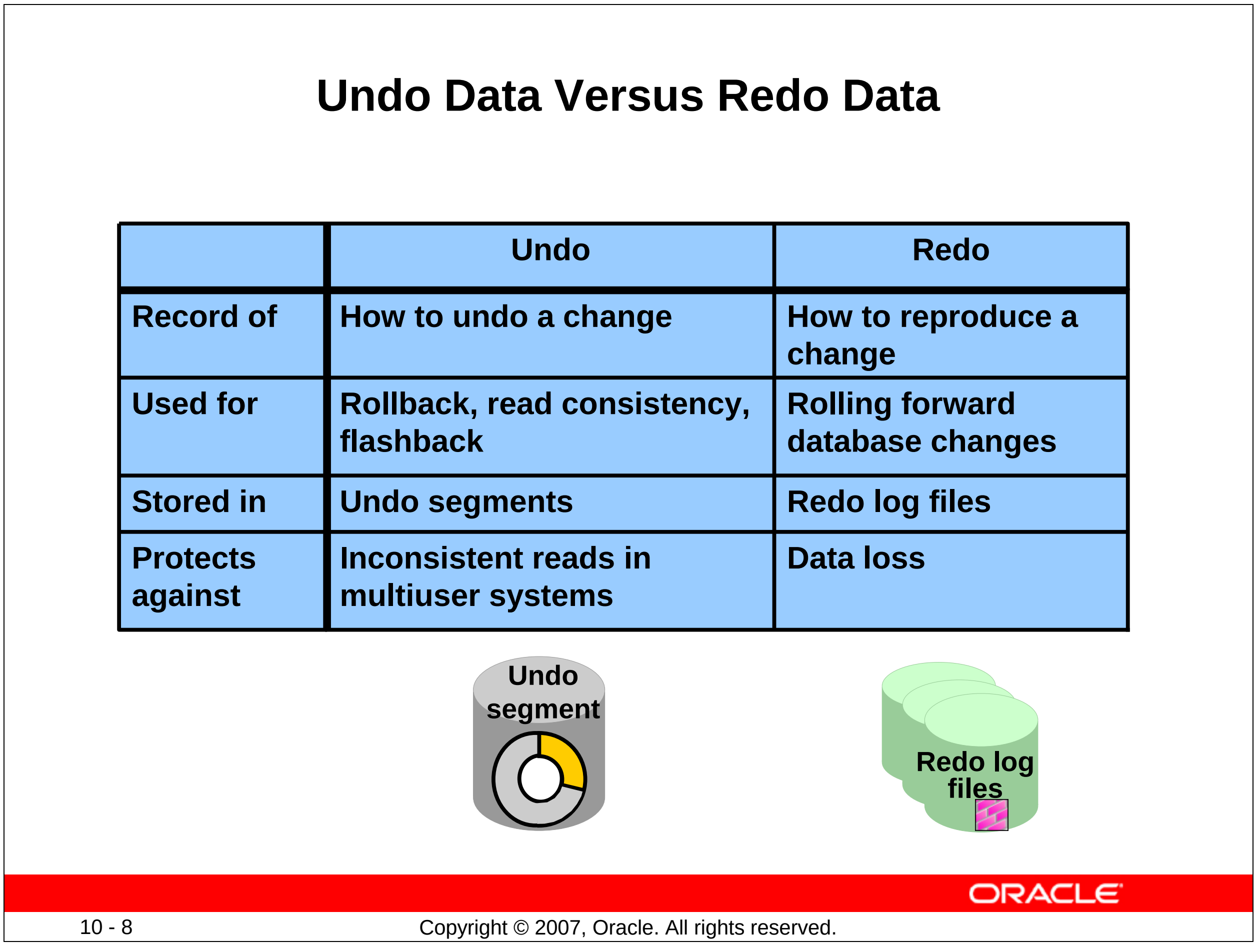
The installation process automatically creates a “smallfile” undo tablespace. You can also create a “bigfile” undo tablespace. However, in a high-volume online transaction processing (OLTP) environment with many short concurrent transactions, contention could occur on the file header. An undo tablespace, stored in multiple data files, can resolve this potential issue.

Although a database may have many undo tablespaces, only one of them at a time can be designated as the current undo tablespace for any instance in the database.

Undo segments are automatically created and always owned by SYS. Because the undo segments act as a circular buffer, each segment has a minimum of two extents. The default maximum number of extents depends on the database block size but is very high (32,765 for an 8 KB block size).

Undo tablespaces are permanent, locally managed tablespaces with automatic extent allocation. They are automatically managed by the database.

Because undo data is required to recover from failed transactions (such as those that may occur when an instance crashes), undo tablespaces can be recovered only while the instance is in the MOUNT state. Recovery considerations for undo tablespaces are covered in the lesson titled “Performing Database Recovery.”



Undo Data Versus Redo Data

	Undo	Redo
Record of	How to undo a change	How to reproduce a change
Used for	Rollback, read consistency, flashback	Rolling forward database changes
Stored in	Undo segments	Redo log files
Protects against	Inconsistent reads in multiuser systems	Data loss

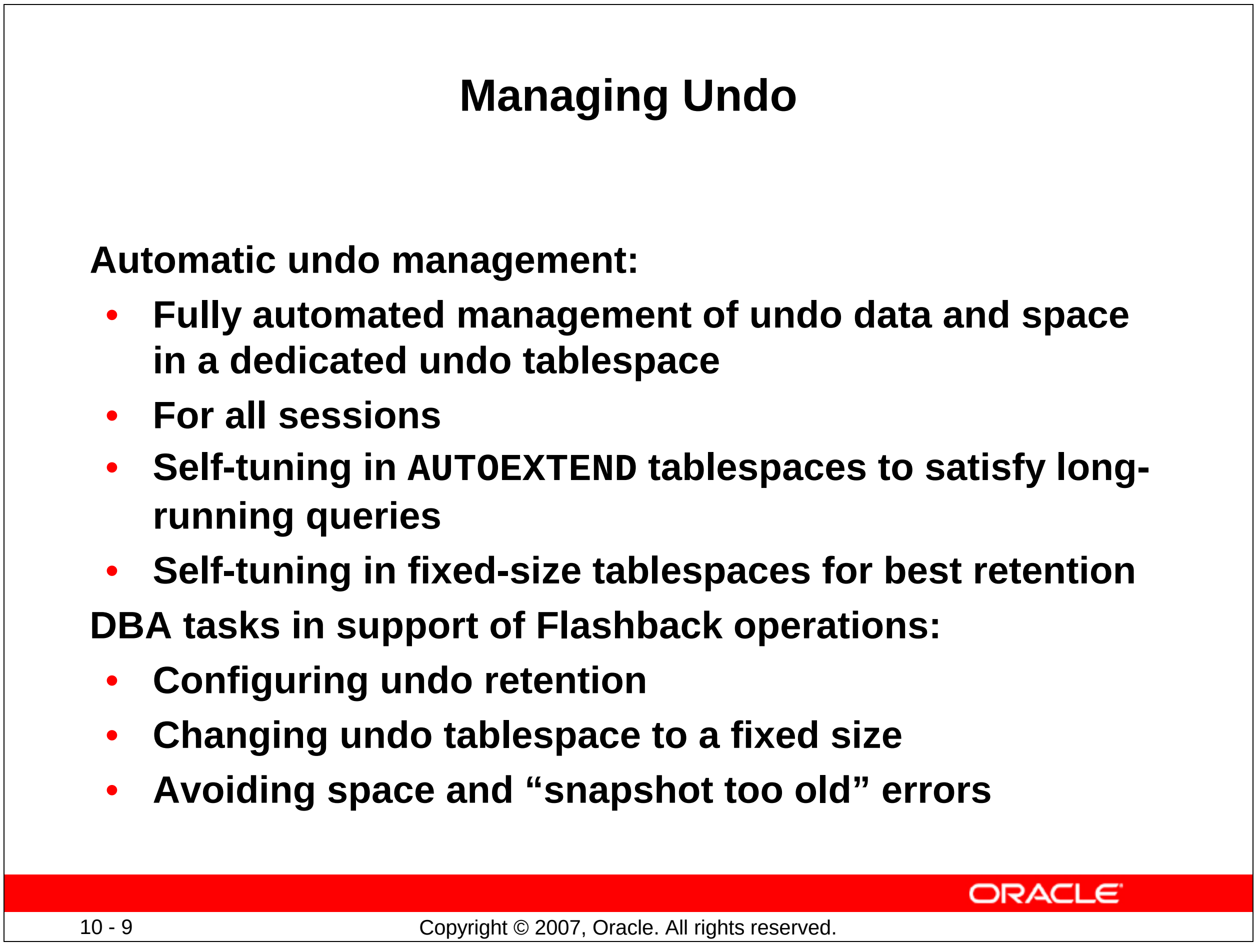
Undo segment
Redo log files
ORACLE

Undo Data Versus Redo Data

Undo data and redo data seem similar at first, but they serve different purposes. Undo data is needed if there is the need to undo a change, and this occurs for read consistency and rollback. Redo data is needed if there is the need to perform the changes again, in cases where they are lost for some reason. Undo block changes are also written to the redo log.

The process of committing entails a verification that the changes in the transaction have been written to the redo log file, which is persistent storage on the disk, as opposed to memory. In addition, the redo log file is typically multiplexed. As a result, there are multiple copies of the redo data on the disk. Although the changes may not yet have been written to the data files where the table’s blocks are actually stored, writing to the persistent redo log is enough to guarantee consistency of the database.

Suppose that a power outage occurs just before committed changes have been reflected in the data files. This situation does not cause a problem because the transaction has been committed. When the system starts up again, it is thus able to roll forward any redo records that are not yet reflected in data files at the time of the outage.



Managing Undo

Automatic undo management:

- Fully automated management of undo data and space in a dedicated undo tablespace
- For all sessions
- Self-tuning in `AUTOEXTEND` tablespaces to satisfy long-running queries
- Self-tuning in fixed-size tablespaces for best retention

DBA tasks in support of Flashback operations:

- Configuring undo retention
- Changing undo tablespace to a fixed size
- Avoiding space and “snapshot too old” errors

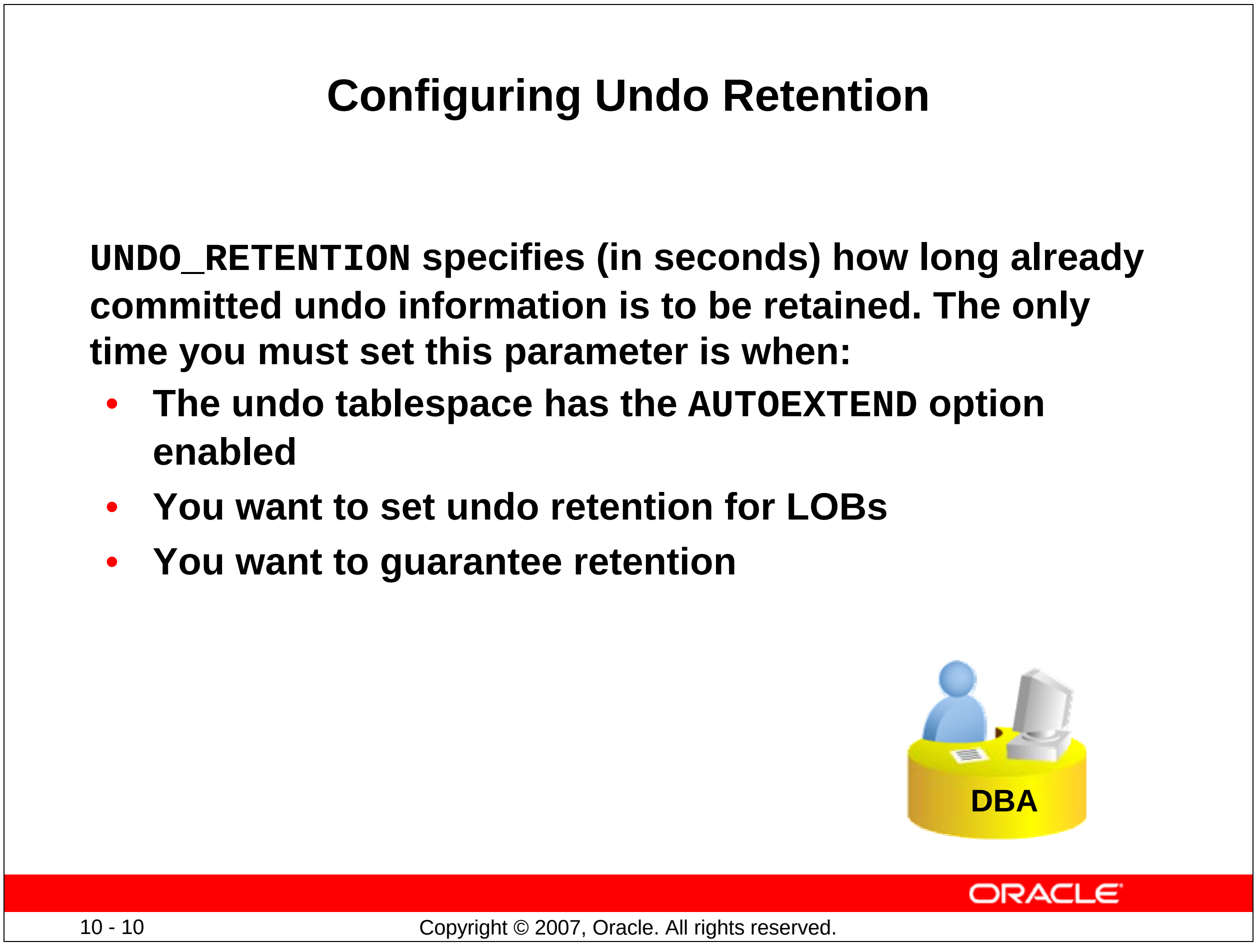
Managing Undo

The Oracle database provides *automatic undo management*, which is a fully automated mechanism for managing undo information and space in a dedicated undo tablespace for all sessions. The system automatically tunes itself to provide the best possible retention of undo information . More precisely, the undo retention period for autoextending tablespaces is tuned to be slightly longer than the longest-running active query. For fixed-size undo tablespaces, the database dynamically tunes for best possible retention.

Automatic undo management is the default for Oracle Database 11*g* (and later releases). Manual undo management is supported for backward compatibility with Oracle8*i* and earlier releases but requires more DBA interaction. In manual undo management mode, undo space is managed through rollback segments (not through undo tablespace).

Note: Oracle strongly recommends that you use automatic undo management.

Although by default the Oracle database manages undo data and space automatically, you may need to perform some tasks if your database is using Flashback operations. The administration of undo should prevent space errors, the use of too much space, and “Snapshot too old” errors.



Configuring Undo Retention

`UNDO_RETENTION` **specifies (in seconds) how long already committed undo information is to be retained. The only time you must set this parameter is when:**

- **The undo tablespace has the `AUTOEXTEND` option enabled**
- **You want to set undo retention for LOBs**
- **You want to guarantee retention**

10 - 10

Configuring Undo Retention

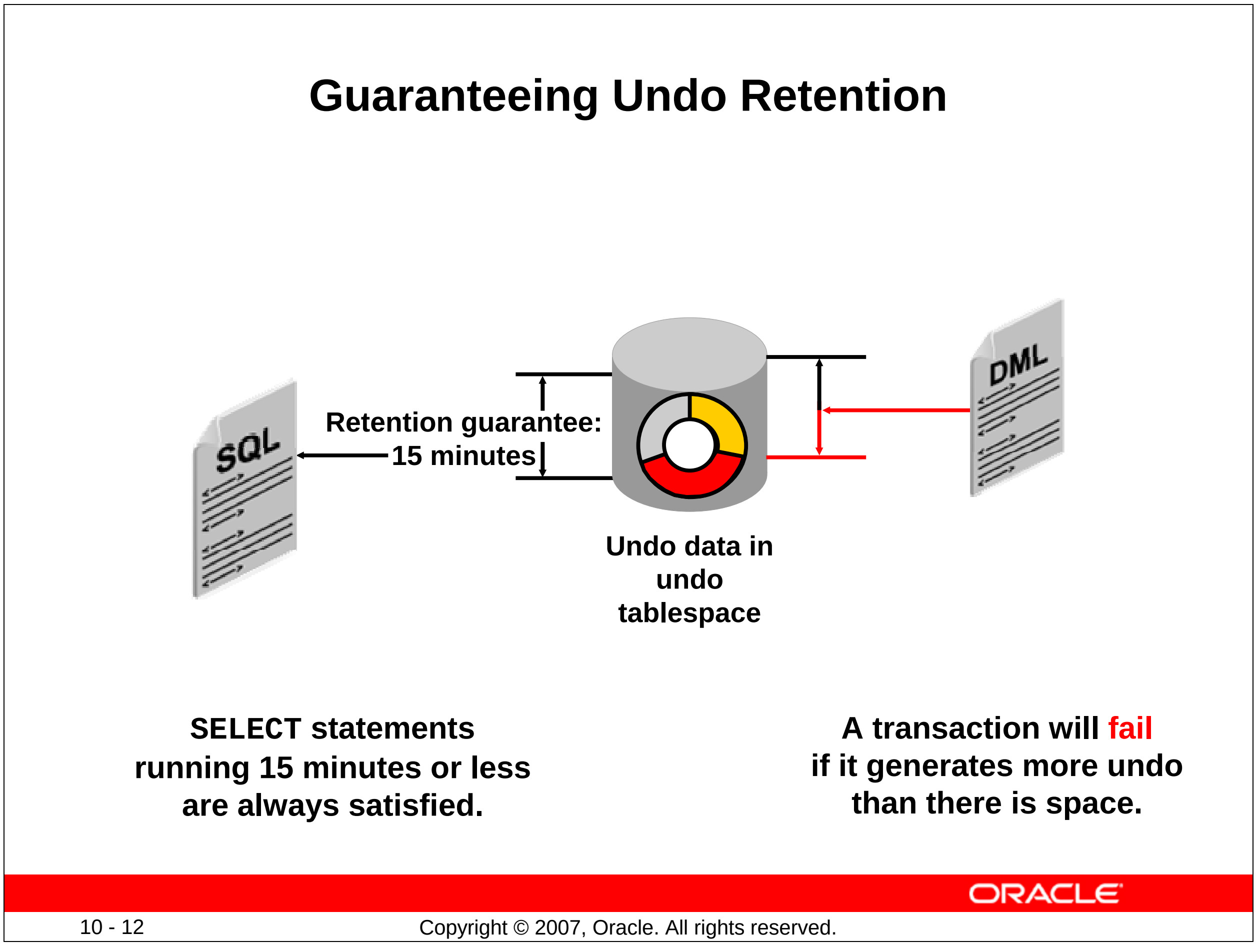
The `UNDO_RETENTION` initialization parameter specifies (in seconds) the low threshold value of undo retention. Set the minimum undo retention period for the autoextending undo tablespace to be as long as the longest expected Flashback operation. For autoextending undo tablespaces, the system retains undo for at least the time specified in this parameter, and automatically tunes the undo retention period to meet the undo requirements of the queries. But this autotuned retention period may be insufficient for your Flashback operations.

For fixed-size undo tablespaces, the system automatically tunes for the best possible undo retention period on the basis of undo tablespace size and usage history; it ignores `UNDO_RETENTION` unless retention guarantee is enabled. So for automatic undo management, the `UNDO_RETENTION` setting is used for the three cases listed in the slide. In cases other than these three, this parameter is ignored.

Configuring Undo Retention (continued)

Undo information is divided into three categories:

- **Uncommitted undo information:** Supports a currently running transaction; is required if a user wants to roll back or if the transaction has failed. Uncommitted undo information is never overwritten.
- **Committed undo information:** Is no longer needed to support a running transaction but is still needed to meet the undo retention interval. It is also known as “unexpired” undo information. Committed undo information is retained when possible without causing an active transaction to fail because of lack of space.
- **Expired undo information:** Is no longer needed to support a running transaction. Expired undo information is overwritten when space is required by an active transaction.



Guaranteeing Undo Retention

Retention guarantee:
15 minutes

Undo data in
undo
tablespace

SELECT statements
running 15 minutes or less
are always satisfied.

A transaction will fail
if it generates more undo
than there is space.

Guaranteeing Undo Retention

The default undo behavior is to overwrite committed transactions that have not yet expired rather than to allow an active transaction to fail because of lack of undo space.

This behavior can be changed by guaranteeing retention. With guaranteed retention, undo retention settings are enforced even if they cause transactions to fail.

`RETENTION GUARANTEE` is a tablespace attribute rather than an initialization parameter. This attribute can be changed only with SQL command-line statements. The syntax to change an undo tablespace to guarantee retention is:

```
SQL> ALTER TABLESPACE undotbs1 RETENTION GUARANTEE;
```

To return a guaranteed undo tablespace to its normal setting, use the following command:

```
SQL> ALTER TABLESPACE undotbs1 RETENTION NOGUARANTEE;
```

The retention guarantee applies only to undo tablespaces. Attempts to set it on a non-undo tablespace result in the following error:

```
SQL> ALTER TABLESPACE example RETENTION GUARANTEE;
ERROR at line 1:
ORA-30044: 'Retention' can only specified for undo
tablespace
```

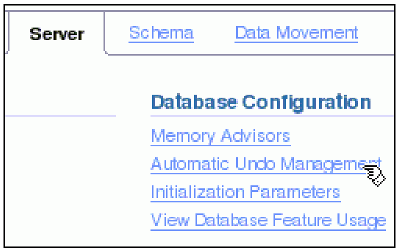
Changing an Undo Tablespace to a Fixed Size

Reasons:

- **Supporting Flashback operations**
- **Limiting tablespace growth**

Workflow:

1. **Run regular workload.**
2. **Self-tuning mechanism establishes minimum required size.**
3. **(Optional) Use Undo Advisor, which calculates required size for future growth.**
4. **(Optional) Change undo tablespace to a fixed size.**


Server
Schema
Data Movement
Database Configuration
Memory Advisors
Automatic Undo Management
Initialization Parameters
View Database Feature Usage

Changing an Undo Tablespace to a Fixed Size

You might have two reasons for changing the undo tablespace to a fixed size: to support Flashback operations (where you expect future use of the undo) or to prevent the tablespace from growing too large.

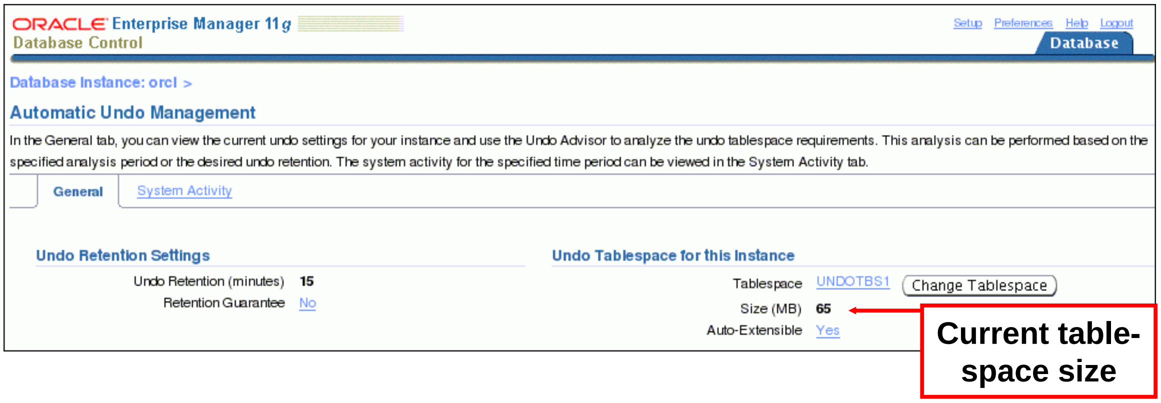
If you decide to change the undo tablespace to a fixed size, you must choose a large enough size to avoid the following two errors:

- DML failures (because there is not enough space to the undo for new transactions)
- "Snapshot too old" errors (because there was insufficient undo data for read consistency)

Oracle recommends that you run a regular, full workload, allowing the undo tablespace to grow to its minimum required size. The automatically gathered statistics include the duration of the longest-running query and the undo generation rate. Computing the minimum undo tablespace size based on these statistics is advisable for a system without Flashback operations, and for a system for which you do not expect longer-running queries in the future.

You can use the Undo Advisor to enter your desired duration for the undo period for longer-running queries and flashback.

General Undo Information



Database Instance: orcl >

Automatic Undo Management

In the General tab, you can view the current undo settings for your instance and use the Undo Advisor to analyze the undo tablespace requirements. This analysis can be performed based on the specified analysis period or the desired undo retention. The system activity for the specified time period can be viewed in the System Activity tab.

Undo Retention Settings

Undo Retention (minutes)	15
Retention Guarantee	No

Undo Tablespace for this Instance

Tablespace	UNDOTBS1 (Change Tablespace)
Size (MB)	65
Auto-Extensible	Yes

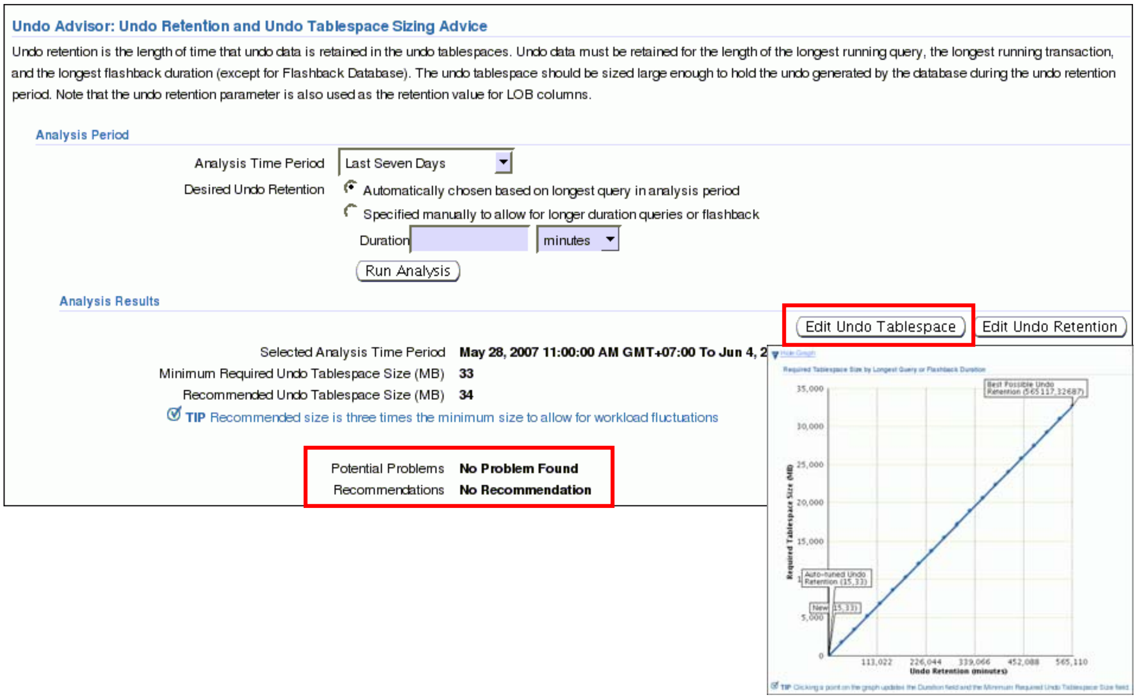
Current table-space size

General Undo Information

In Enterprise Manager, select Server > Automatic Undo Management.

There are two pages: General and System Activity. In the top part of the General page, you see the Undo Retention Settings and information about the undo tablespace for this instance.

Using the Undo Advisor



Undo Advisor: Undo Retention and Undo Tablespace Sizing Advice

Undo retention is the length of time that undo data is retained in the undo tablespaces. Undo data must be retained for the length of the longest running query, the longest running transaction, and the longest flashback duration (except for Flashback Database). The undo tablespace should be sized large enough to hold the undo generated by the database during the undo retention period. Note that the undo retention parameter is also used as the retention value for LOB columns.

Analysis Period

Analysis Time Period: Last Seven Days

Desired Undo Retention:
● Automatically chosen based on longest query in analysis period
○ Specified manually to allow for longer duration queries or flashback

Duration: minutes

Analysis Results

Selected Analysis Time Period: May 28, 2007 11:00:00 AM GMT+07:00 To Jun 4, 2

Minimum Required Undo Tablespace Size (MB): 33

Recommended Undo Tablespace Size (MB): 34

TIP Recommended size is three times the minimum size to allow for workload fluctuations

Potential Problems: No Problem Found

Recommendations: No Recommendation

TIP Clicking a point on the graph updates the Duration field and the Minimum Required Undo Tablespace Size field.

Run Analysis
Edit Undo Tablespace
Edit Undo Retention
Hide Graph
Required Tablespace Size by Longest Query or Flashback Duration
Required Tablespace Size (MB)
35,000
30,000
25,000
20,000
15,000
5,000
0
113,022
226,044
339,066
452,088
565,110
Undo Retention (minutes)
Best Possible Undo Retention (565117,32687)
Auto-tuned Undo Retention (15,33)
New (15,33)

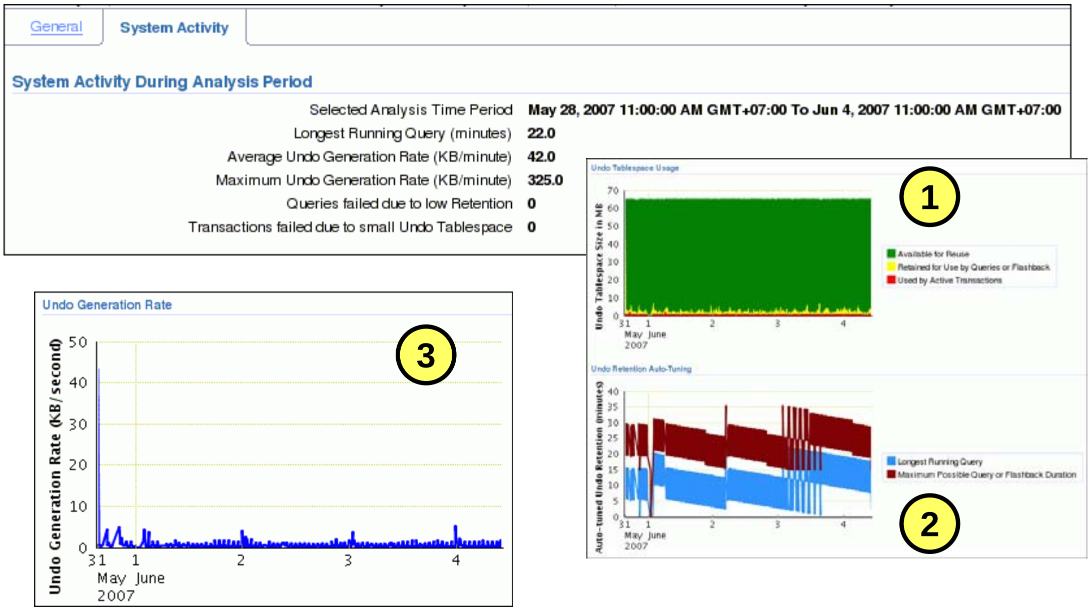
Using the Undo Advisor

The middle part of the General Undo page is your access to the Undo Advisor. It provides an estimate of the undo tablespace size required to satisfy a given undo retention.

The analysis region of the advisor displays the tablespace size required to support the retention period. You can click a point on the graph to see the tablespace size required to support the selected period.

Click the Edit Undo Tablespace button, and then click Edit in the Datafile section to change the undo tablespace to a fixed size.

Viewing System Activity



System Activity During Analysis Period

Selected Analysis Time Period	May 28, 2007 11:00:00 AM GMT+07:00 To Jun 4, 2007 11:00:00 AM GMT+07:00
Longest Running Query (minutes)	22.0
Average Undo Generation Rate (KB/minute)	42.0
Maximum Undo Generation Rate (KB/minute)	325.0
Queries failed due to low Retention	0
Transactions failed due to small Undo Tablespace	0

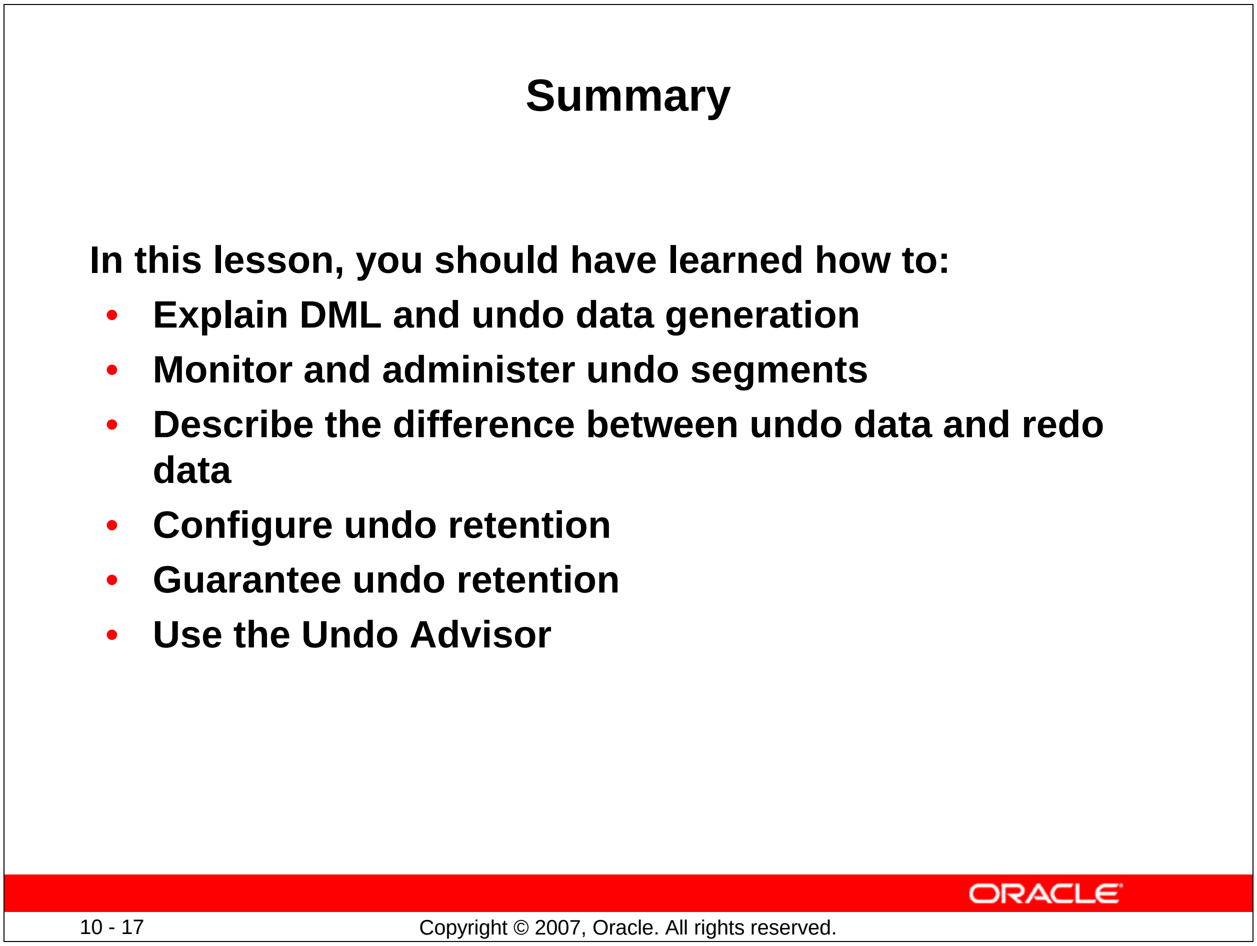
Undo Tablespace Usage
Undo Tablespace Size in MB
70
60
50
40
30
20
10
0
31
1
2
3
4
May June
2007
Available for Reuse
Retained for Use by Queries or Flashback
Used by Active Transactions
1
Undo Retention Auto-Tuning
Auto-tuned Undo Retention (minutes)
40
35
30
25
20
15
10
5
0
31
1
2
3
4
May June
2007
Longest Running Query
Maximum Possible Query or Flashback Duration
2
Undo Generation Rate
Undo Generation Rate (KB/second)
50
40
30
20
10
0
31
1
2
3
4
May June
2007
3

Viewing System Activity

The top part of the page displays system activity during the selected period.

Beneath this, there are three graphs:

1. **Undo Tablespace Usage:** Shows the tablespace size (in MB) by days of the month
2. **Undo Retention Auto-Tuning:** Visualizes the tuned undo retention (in minutes) by days of the month
3. **Undo Generation Rate:** Displays the undo generation (in KB per seconds) by days of the month

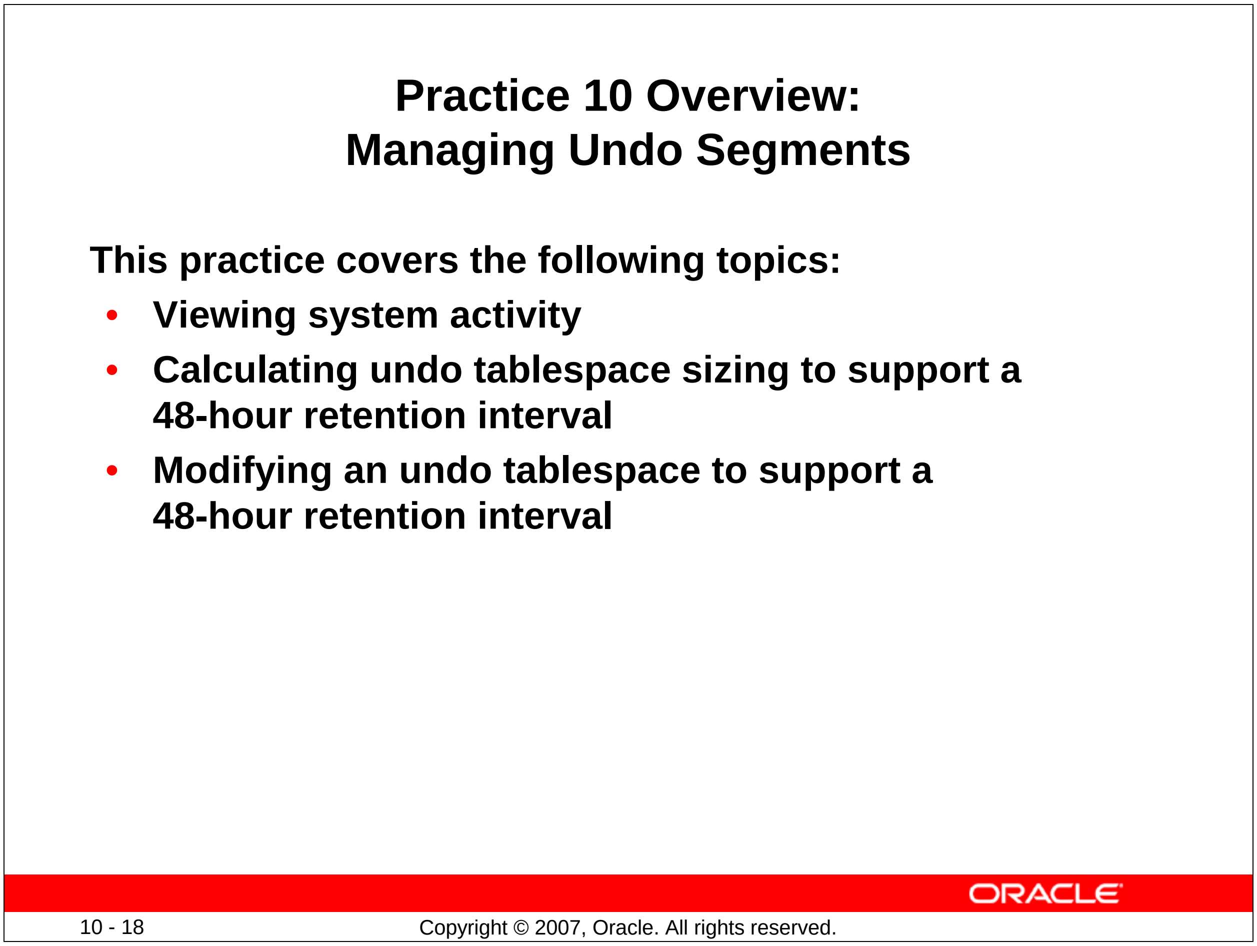


Summary

In this lesson, you should have learned how to:

- Explain DML and undo data generation
- Monitor and administer undo segments
- Describe the difference between undo data and redo data
- Configure undo retention
- Guarantee undo retention
- Use the Undo Advisor

10 - 17



Practice 10 Overview: Managing Undo Segments

This practice covers the following topics:

- Viewing system activity
- Calculating undo tablespace sizing to support a 48-hour retention interval
- Modifying an undo tablespace to support a 48-hour retention interval

11

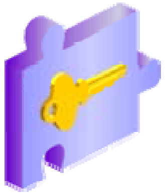
Implementing Oracle Database Security

ORACLE

Objectives

After completing this lesson, you should be able to:

- **Describe DBA responsibilities for security**
- **Apply the principle of least privilege**
- **Enable standard database auditing**
- **Specify audit options**
- **Review audit information**
- **Maintain the audit trail**



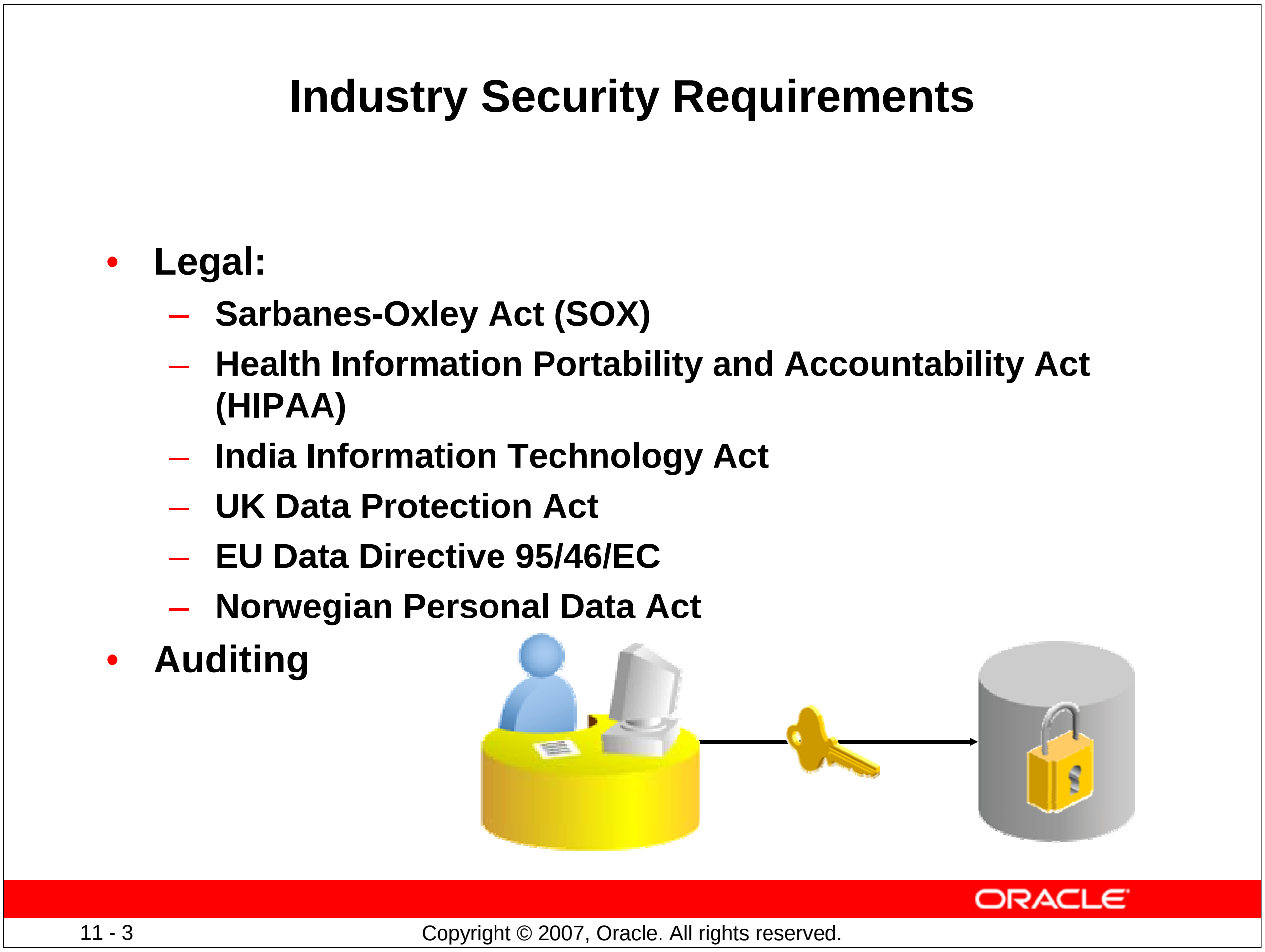
Objectives

This lesson is a starting point for learning about Oracle Security. Additional information is provided in the following documentation:

- *Oracle Database Concepts 11g Release 1 (11.1)*
- *Oracle Database Administrator's Guide 11g Release 1 (11.1)*
- *Oracle Database Security Guide 11g Release 1 (11.1)*

Additional training is provided in the following courses:

- *Oracle Database 11g: Administration Workshop II* (D50079GC10)
- *Oracle Database 11g: Security*



Industry Security Requirements

- Legal:
  - Sarbanes-Oxley Act (SOX)
  - Health Information Portability and Accountability Act (HIPAA)
  - India Information Technology Act
  - UK Data Protection Act
  - EU Data Directive 95/46/EC
  - Norwegian Personal Data Act
- Auditing

11 - 3

Industry Security Requirements

Security requirements have been a matter of individual concern until recently. Unless you were handling government or military data, there were few legal requirements. This is rapidly changing. A variety of laws have been passed to enforce the privacy and accuracy of data. Along with these laws, there is a requirement to audit the security measures that are in place. These laws vary by country, but the concerns are the same and many of the solutions are the same.

Legal: Each of the laws listed here has specific requirements. This list is representative of many other laws that are being passed worldwide. Of course, security laws vary from place to place.

- **Sarbanes-Oxley Act (SOX)** requires that public companies strengthen and document internal controls to prevent individuals from committing fraudulent acts that may compromise an organization’s financial position or the accuracy of its financial statements. The chief executive officer and the chief financial officer must attest to the adequacy of the internal controls and accuracy of the financial report. These officers are subject to fines and imprisonment for fraudulent reports. The details of SOX include requirements for providing the information that is used to generate the reports, as well as internal controls that are used to assure the integrity of the financial information.

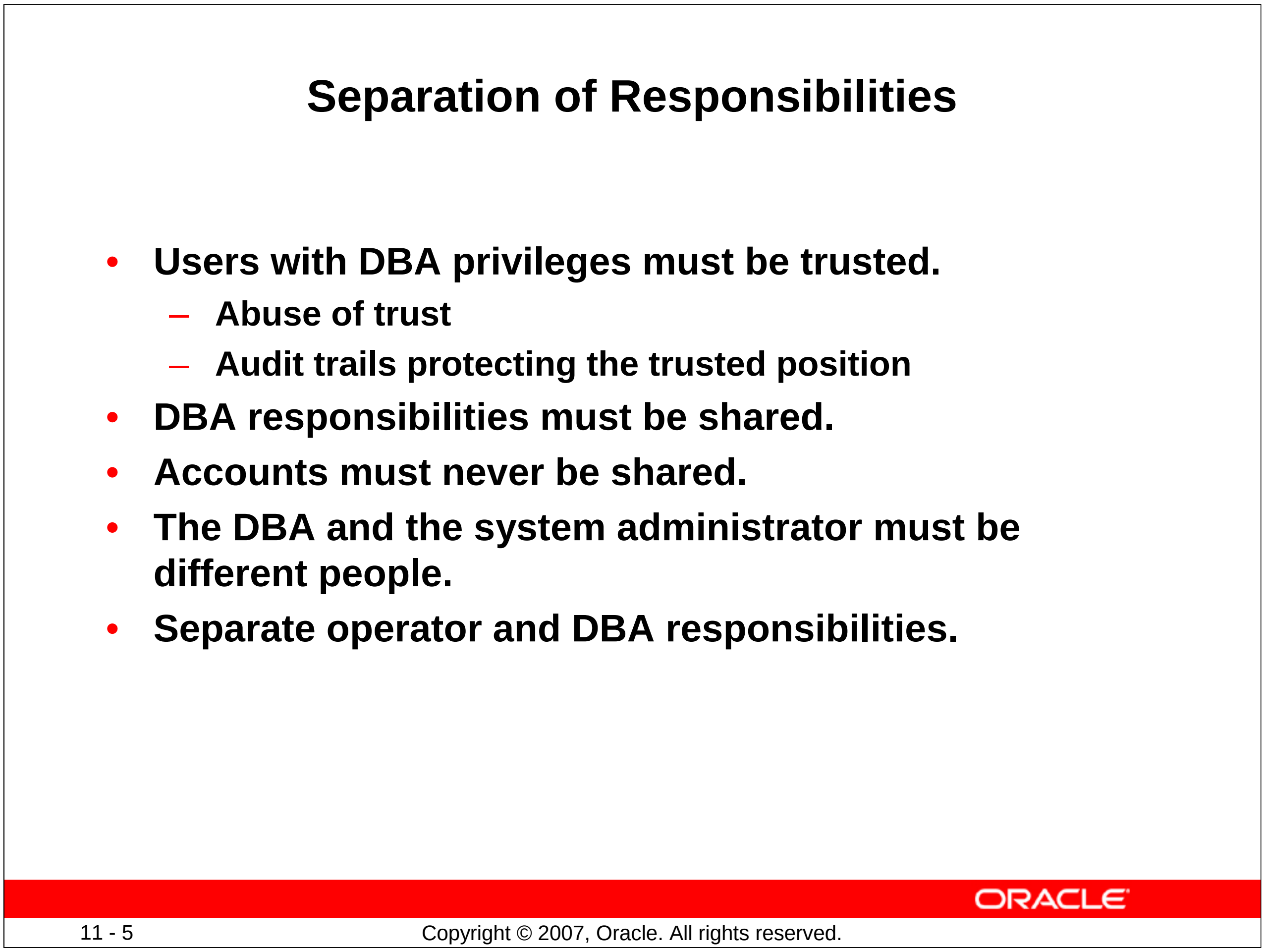
Industry Security Requirements (continued)

- **Health Information Portability and Accountability Act (HIPAA)** is intended to protect personally identifiable health information from release or misuse. Information holders must provide audit trails of all who access this data.
- **EU Data Directive** is intended to protect individual privacy and sets a standard across all of the member countries of the European Union.
- **UK Data Protection Act** is intended to protect individual privacy by restricted access to individually identifiable data. It has eight points, one of which requires that data be kept secure.
- **Norwegian Personal Data Act** is compatible with and goes beyond the EU data directive in some areas.
- **Other laws:**
  - **Family Educational Rights and Privacy Act (FERPA)** covers health and personal information held by schools.
  - **California Breach Law** requires that an organization holding a variety of personal identity information (PII) (for example, credit card, driver's license, and government identity numbers) must protect that information. If the information has been compromised, the organization must notify all individuals involved. There are two laws, CA-SB-1386 and CA-AB-1950, that apply to organizations that hold PII.
  - **Federal Information Security Management Act (FISMA)** is creating security guidance and standards through Federal Information Processing Standard (FIPS) documents that are managed by the National Institute of Standards (NIST). These standards are applied to organizations that are processing information for the U.S. government.
  - **Payment Card Industry (PCI) data security standard** has become law in Minnesota; other states are considering similar measures. The PCI standard is enforced worldwide by contract.

Auditing: Many of these laws include provisions requiring that security plans (internal controls) be audited periodically. SOX requirements are vague and subject to interpretation by the officers of the organization. The implementation details can vary widely, depending on the level of details that the officers require. Because SOX is vague but has severe penalties, it is important to protect your company. The cost of security measures must be balanced against the risk. No one will certify that you are 100% secure. A very good solution is industry consensus. If you meet the agreed-upon minimum security practices and have accomplished due diligence, you may be safe from the worst penalties of the law. Some good resources for industry-standard practices are the SANS Institute (SANS = SysAdmin, Audit, Network, Security), CERT/CC (⫽⫽⌜✕✓▲⌜⌟□✓⋱□□✓✕●⌜■•⌜□□⌜◜◜⫽●□□●•⬍⌜✕?•▲⋱□⌟⫽✕□ ▲•⌜□□⌜⫽✓✕▲◞⌜●▲□⫽⌟□□⌜⌟⌜●?⌜□□□✓●⌟□▲•⌜□□□□□▭⏷⏷▾▾□ ⌜⌜✕▲●⌟•⌜✓▲•⫽●□?▲✓●⌟✓✕⌟□

- http://www.sans.org/index.php
- http://www.cert.org/nav/index.html
- http://www.iso17799software.com/

The ISO-17799 ⌜⌜✕▲•⌟•⌜✓▲•⫽●□?▲✓●⌟✓✕⌟ is an international standard of security practices. It includes best practices, certification, and risk assessment. It covers a broad range of issues and includes prewritten policies.



Separation of Responsibilities

- Users with DBA privileges must be trusted.
  - Abuse of trust
  - Audit trails protecting the trusted position
- DBA responsibilities must be shared.
- Accounts must never be shared.
- The DBA and the system administrator must be different people.
- Separate operator and DBA responsibilities.

Separation of Responsibilities

These are the main requirements to satisfy the separation of duties.

DBAs must be trusted: It is difficult to restrict a DBA. To do his or her job, the DBA requires high-level privileges. A DBA has a position of trust and must be thoroughly vetted. Even a trusted DBA must have accountability. Consider the following:

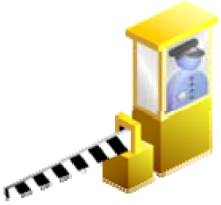
- **Abuse of trust:** A DBA can potentially misuse the encrypted passwords from the `DBA_USERS` view.
- **Audit trails protecting the trusted position:** When auditing is carefully implemented and guidelines have been followed, the audit trail can show that a particular person has not violated procedures or committed a damaging act. If a malicious user tries to cast suspicion on a trusted user, well-designed audit trails catch the act.

Oracle Database Vault: The Oracle Database Vault option can be purchased for situations in which the separation of duties must be enforced by the database, or for situations in which the DBA is not allowed to view data in some or all database schemas.

Database Security

A secure system ensures the confidentiality of the data that it contains. There are several aspects of security:

- **Restricting access to data and services**
- **Authenticating users**
- **Monitoring for suspicious activity**



ORACLE

11 - 6

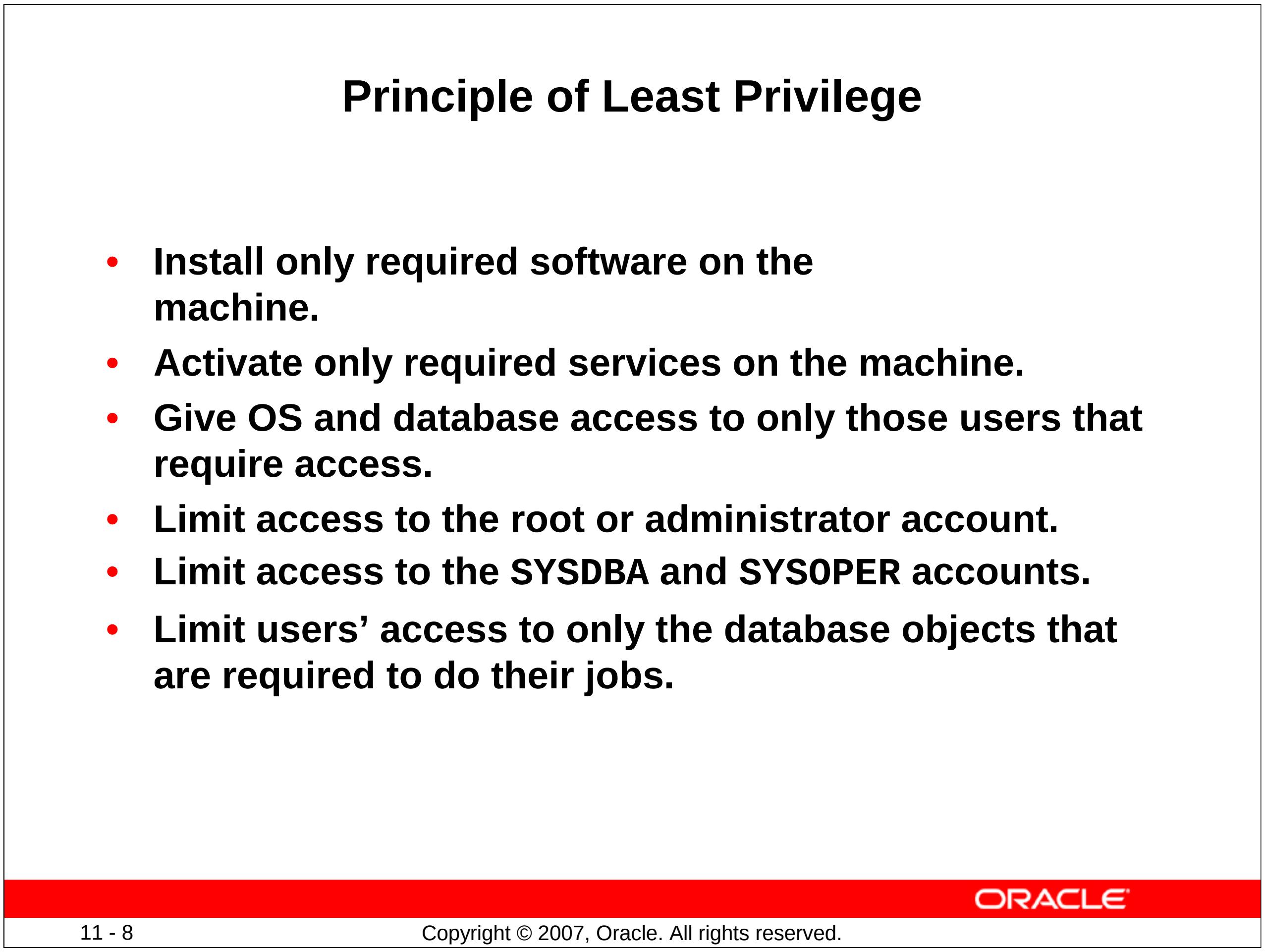
Database Security

Oracle Database 11*g* provides the industry’s best framework for a secure system. But for that framework to be effective, the database administrator must follow best practices and continually monitor database activity.

- **Restricting Access to Data and Services**
  All users must not have access to all data. Depending on what is stored in your database, restricted access can be mandated by business requirements, by customer expectations, and (increasingly) by legal restrictions. Credit card information, health-care data, identity information, and so on must be protected from unauthorized access. The Oracle database provides extremely fine-grained authorization controls to limit database access. Restricting access must include applying the principle of least privilege.

Database Security (continued)

- **Authenticating Users**
  To enforce access controls on sensitive data, the system must first know who is trying to access the data. Compromised authentication can render all other security precautions useless. The most basic form of user authentication is challenging users to provide something that they know, such as a password. Ensuring that passwords follow simple rules can greatly increase the security of your system. Stronger authentication methods include requiring users to provide something that they have, such as a token or public key infrastructure (PKI) certificate. An even stronger form of authentication is to identify users through a unique biometric characteristic such as a fingerprint, an iris scan, bone structure patterns, and so on. The Oracle database supports advanced authentication techniques (such as token-, biometric-, and certificate-based identification) through the Advanced Security option. User accounts that are not in use must be locked to prevent attempts to compromise authentication.
- **Monitoring for Suspicious Activity**
  Even authorized and authenticated users can sometimes compromise your system. Identifying unusual database activity (such as an employee who suddenly begins querying large amounts of credit card information, research results, or other sensitive information) can be the first step to detecting information theft. The Oracle database provides a rich set of auditing tools to track user activity and identify suspicious trends.



Principle of Least Privilege

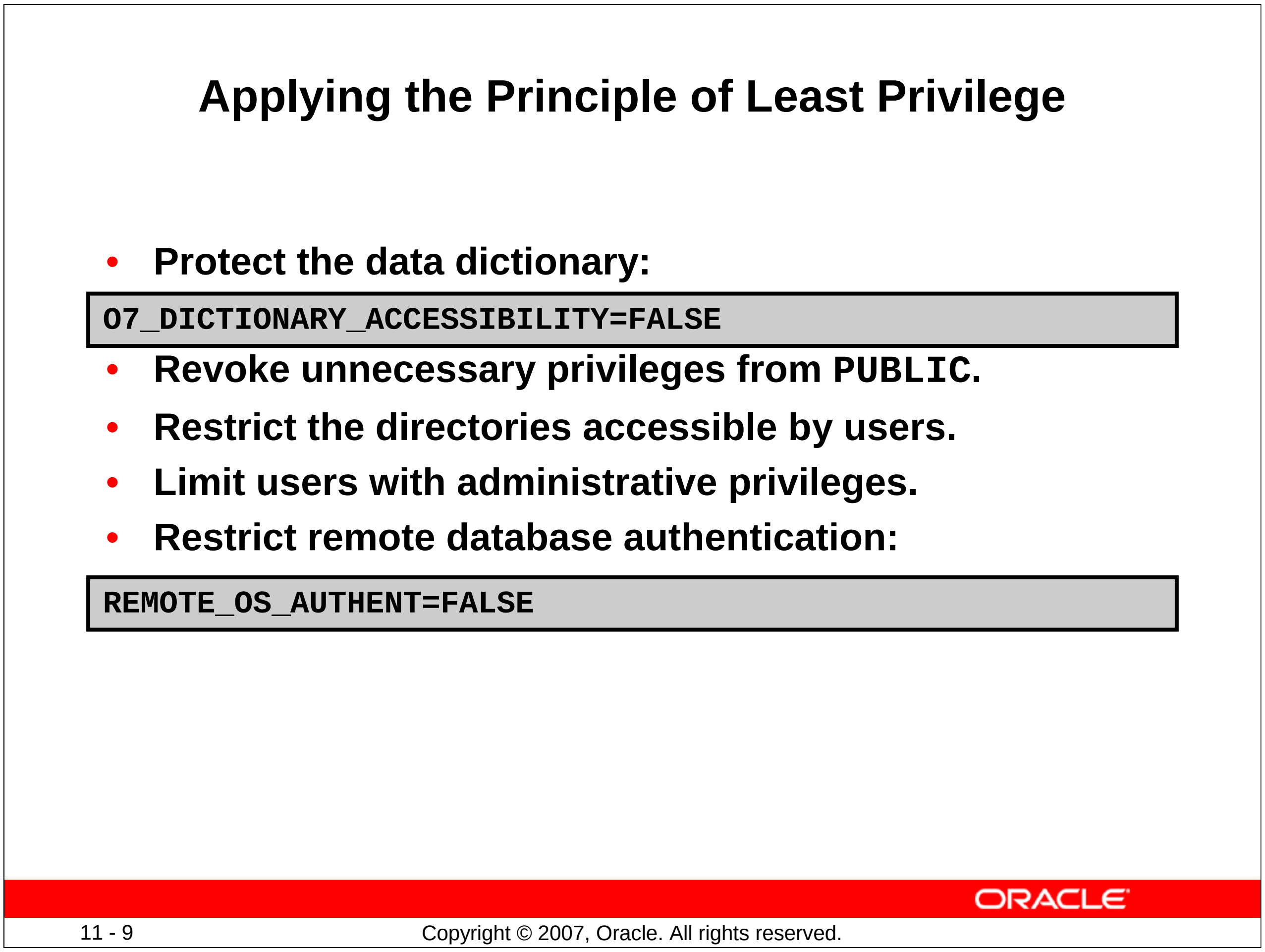
- Install only required software on the machine.
- Activate only required services on the machine.
- Give OS and database access to only those users that require access.
- Limit access to the root or administrator account.
- Limit access to the SYSDBA and SYSOPER accounts.
- Limit users’ access to only the database objects that are required to do their jobs.

11 - 8

Principle of Least Privilege

Apply the principle of least privilege, starting at the lowest levels and continuing at every level. There are always new security exploits that cannot be anticipated. By applying this principle, the possibility of the exploit is reduced and the damage may be contained.

- **Install only required software on the machine:** By reducing the number of software packages, you reduce maintenance, upgrades, the possibility of security holes, and software conflicts.
- **Activate only required services on the machine:** Fewer services imply fewer open ports and fewer attack vectors.
- **Give operating system (OS) and database access to only those users that require access:** Fewer users mean fewer passwords and accounts. This reduces the possibility of open or stale accounts. Fewer accounts make it easier for the administrator to keep the accounts current.
- **Limit access to the root or administrator account:** The administrator account must be carefully guarded, audited, and never shared.
- **Limit access to the SYSDBA and SYSOPER accounts:** Users who require access to these roles must each have their own account and be audited.
- **Limit users’ access to only the database objects required to do their jobs:** Users who have access to more objects and services than they require have an opportunity for mischief.



Applying the Principle of Least Privilege

- Protect the data dictionary:

```
O7_DICTIONARY_ACCESSIBILITY=FALSE
```

- Revoke unnecessary privileges from `PUBLIC`.
- Restrict the directories accessible by users.
- Limit users with administrative privileges.
- Restrict remote database authentication:

```
REMOTE_OS_AUTHENT=FALSE
```

Applying the Principle of Least Privilege

The principle of least privilege means that a user must be given only those privileges that are required to efficiently complete a task. This reduces the chances of users modifying or viewing data (either accidentally or maliciously) that they do not have the privilege to modify or view.

Protect the data dictionary: The `O7_DICTIONARY_ACCESSIBILITY` parameter is set by default to `FALSE`. You must not allow this to be changed without a very good reason because it prevents users with the `ANY TABLE` system privileges from accessing the data dictionary base tables. It also ensures that the `SYS` user can log in only as `SYSDBA`.

Revoke unnecessary privileges from `PUBLIC`: Several packages are extremely useful to applications that need them, but require proper configuration to be used securely. `PUBLIC` is granted execute privilege on the following packages: `UTL_SMTP`, `UTL_TCP`, `UTL_HTTP`, and `UTL_FILE`. In Oracle Database 11*g*, network access is controlled by an access control list (ACL) that may be configured to allow certain users access to specific network services. Network access is denied by default. An ACL must be created to allow network access. File through `UTL_FILE` access is controlled at two levels: at the OS level with permissions on files and directories, and in the database by `DIRECTORY` objects that allow access to specific file system directories.The `DIRECTORY` object may be granted to a user for read or for read and write. Execute privileges on other PL/SQL packages should be carefully controlled.

Applying the Principle of Least Privilege (continued)

The more powerful packages that may potentially be misused include:

- **UTL_SMTP:** Permits arbitrary email messages to be sent by using the database as a Simple Mail Transfer Protocol (SMTP) mail server. Use the ACL to control which machines may be accessed by which users.
- **UTL_TCP:** Permits outgoing network connections to be established by the database server to any receiving or waiting network service. Thus, arbitrary data can be sent between the database server and any waiting network service. Use the ACL to control access.
- **UTL_HTTP:** Allows the database server to request and retrieve data via HTTP. Granting this package to a user may permit data to be sent via HTML forms to a malicious Web site. Limit access by using the ACL.
- **UTL_FILE:** If configured improperly, allows text-level access to any file on the host operating system. When properly configured, this package limits user access to specific directory locations.

Restrict access to OS directories: The DIRECTORY object inside the database enables DBAs to map directories to OS paths and to grant privileges on those directories to individual users.

Limit users with administrative privileges: Do not provide database users more privileges than necessary. Nonadministrators must not be granted the DBA role. To implement least privilege, restrict the following types of privileges:

- Grants of system and object privileges
- SYS-privileged connections to the database, such as SYSDBA and SYSOPER
- Other DBA-type privileges, such as DROP ANY TABLE

Restrict remote database authentication: The REMOTE_OS_AUTHENT parameter is set to FALSE by default. It must not be changed unless all clients can be trusted to authenticate users appropriately. With the advent of Secure External Password Store (available in Oracle Database 10*g* Release 2), there are few compelling reasons ever to allow remote OS authentication.

In the remote authentication process:

- The database user is authenticated externally
- The remote system authenticates the user
- The user logs in to the database without further authentication

Protect Privileged Accounts

Privileged accounts can be protected by:

- **Using password file with case-sensitive passwords**
- **Enabling strong authentication for administrator roles**
  - **Grant administrator roles in Oracle Internet Directory**
  - **Use Kerberos tickets**
  - **Use certificates with SSL**

Setting Database Administrator Authentication

Users with SYSDBA, SYSOPER, or SYSASM privileges must always be authenticated. When connecting locally, the user is authenticated by the local OS by being a member of a privileged OS group. If connecting remotely, a password file is used to authenticate privileged users. If the password file is configured, it will be checked first. In Oracle Database 11*g*, these passwords are case-sensitive. Oracle Database 11*g* provides other methods that make remote administrator authentication more secure and centralize the administration of these privileged users.

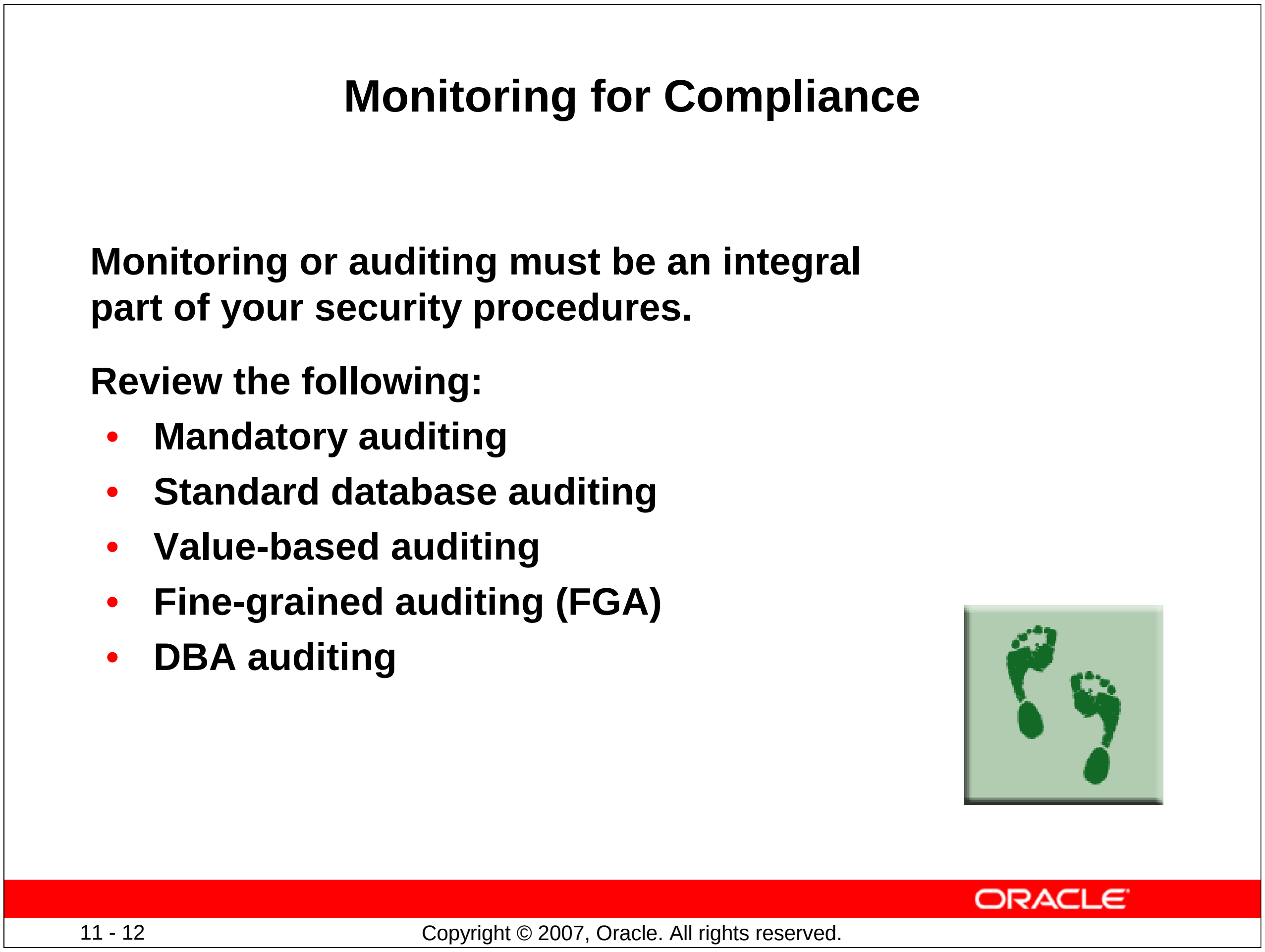
When a database is created using the Database Configuration Assistant, the password file is case-sensitive. If you upgrade from earlier database versions, be sure to make the password file case-sensitive for remote connections:

```
orapwd file=orapworcl entries=5 ignorecase=N
```

If your concern is that the password file might be vulnerable or that the maintenance of many password files is a burden, strong authentication can be implemented:

- Grant OSDBA or OSOPER roles in Oracle Internet Directory.
- Use Kerberos tickets
- Use certificates over SSL

The Advanced Security option is required if you want to use strong authentication methods. For more information about strong authentication, see the *Oracle Database Advanced Security Administrator’s Guide*.



Monitoring for Compliance

Monitoring or auditing must be an integral part of your security procedures.

Review the following:

- **Mandatory auditing**
- **Standard database auditing**
- **Value-based auditing**
- **Fine-grained auditing (FGA)**
- **DBA auditing**

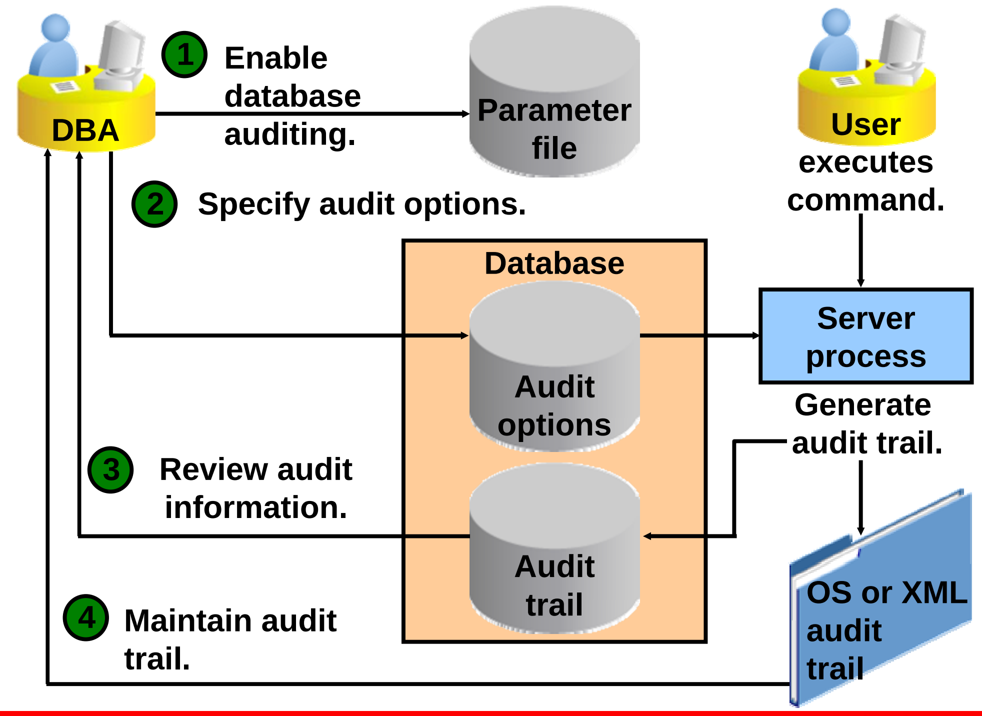
11 - 12

Monitoring for Compliance

Auditing, which means capturing and storing information about what is happening in the system, increases the amount of work the system must do. Auditing must be focused so that only events that are of interest are captured. Properly focused auditing has minimal impact on system performance. Improperly focused auditing can significantly affect performance.

- **Mandatory auditing:** All Oracle databases audit certain actions regardless of other audit options or parameters. The reason for mandatory audit logs is that the database needs to record some database activities, such as connections by privileged users.
- **Standard database auditing:** Enabled at the system level by using the `AUDIT_TRAIL` initialization parameter. After you enable auditing, select the objects and privileges that you want to audit and set the auditing properties with the `AUDIT` command.
- **Value-based auditing:** Extends standard database auditing, capturing not only the audited event that occurred but also the actual values that were inserted, updated, or deleted. Value-based auditing is implemented through database triggers.
- **Fine-grained auditing (FGA):** Extends standard database auditing, capturing the actual SQL statement that was issued rather than only the fact that the event occurred
- **DBA auditing:** Separates the auditing duties between the DBA and an auditor or security administrator who monitors the DBA activities in an operating system audit trail

Standard Database Auditing



DBA

1 Enable database auditing.

Parameter file

User executes command.

2 Specify audit options.

Database

Audit options

Server process

Generate audit trail.

3 Review audit information.

Audit trail

4 Maintain audit trail.

OS or XML audit trail

Standard Database Auditing

After you enable database auditing and specify the auditing options (login events, exercise of system and object privileges, or the use of SQL statements), the database begins collecting audit information.

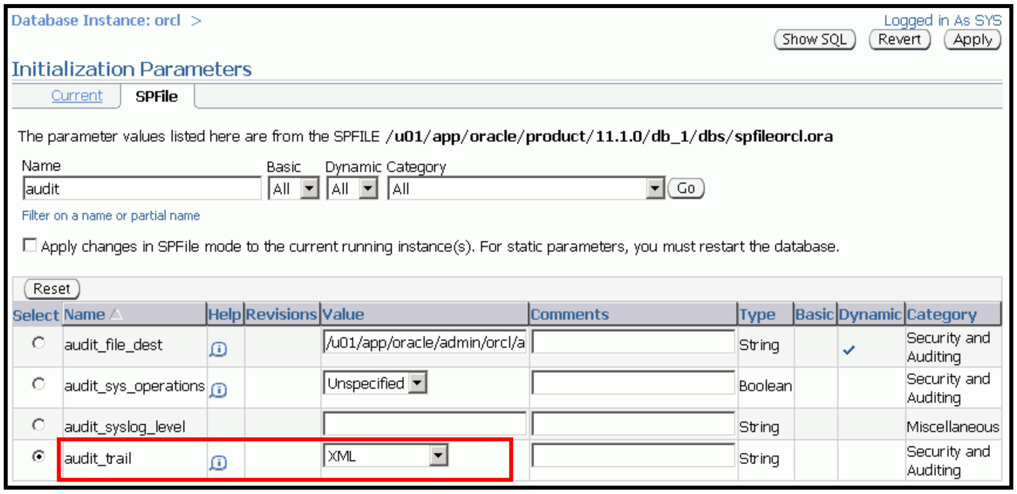
If `AUDIT_TRAIL` is set to `OS`, the audit records are stored in the operating system's audit system. In a Windows environment, this is the event log. In a UNIX or Linux environment, audit records are stored in a file that is specified with the `AUDIT_FILE_DEST` parameter.

If the `AUDIT_TRAIL` parameter is set to `DB`, you can review audit records in the `DBA_AUDIT_TRAIL` view, which is part of the `SYS` schema.

If `AUDIT_TRAIL` is set to `XML` or to `XML,EXTENDED`, the audit records are written to XML files in the directory to which the `AUDIT_FILE_DEST` parameter points. The `V$XML_AUDIT_TRAIL` view allows you to view all the XML files in this directory.

Maintaining the audit trail is an important administrative task. Depending on the focus of the audit options, the audit trail can grow very large very quickly. If not properly maintained, the audit trail can create so many records that it affects the performance of the system. Audit overhead is directly related to the number of records that are produced.

Enabling Auditing



Initialization Parameters

The parameter values listed here are from the SPFILE **/u01/app/oracle/product/11.1.0/db_1/dbs/spfileorcl.ora**

Name: audit Basic: All Dynamic: All Category: All

□ Apply changes in SPFile mode to the current running instance(s). For static parameters, you must restart the database.

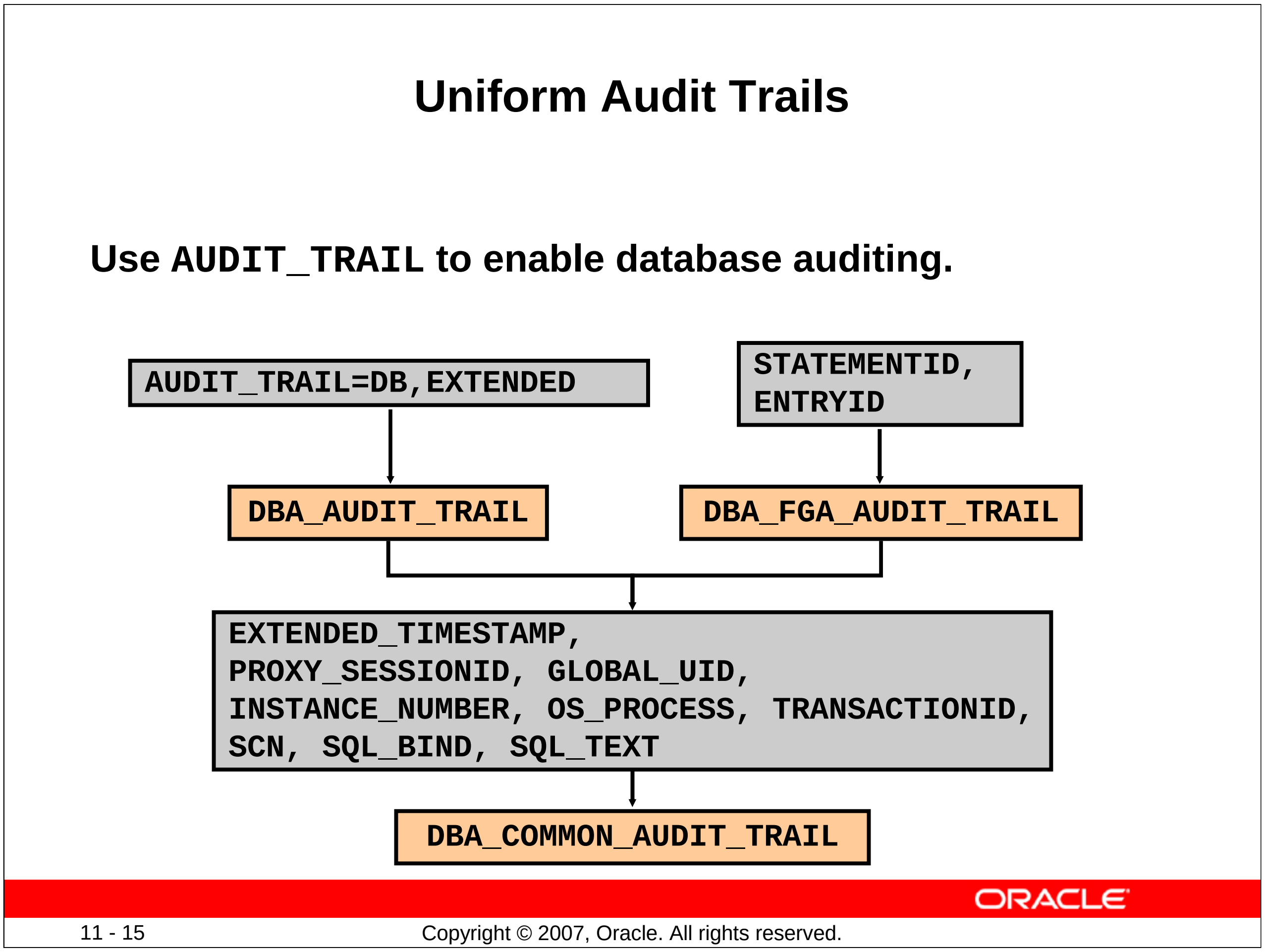
Select	Name △	Help	Revisions	Value	Comments	Type	Basic	Dynamic	Category
○	audit_file_dest	ⓘ		/u01/app/oracle/admin/orcl/a		String		✓	Security and Auditing
○	audit_sys_operations	ⓘ		Unspecified		Boolean			Security and Auditing
○	audit_syslog_level					String			Miscellaneous
●	audit_trail	ⓘ		XML		String			Security and Auditing

```
ALTER SYSTEM SET audit_trail=“XML” SCOPE=SPFILE;
```

Restart database after modifying a static initialization parameter.

Enabling Auditing

You must enable database auditing before audit settings will produce audit records.



Uniform Audit Trails

Use `AUDIT_TRAIL` to enable database auditing.

AUDIT_TRAIL=DB,EXTENDED

STATEMENTID, ENTRYID

DBA_AUDIT_TRAIL

DBA_FGA_AUDIT_TRAIL

EXTENDED_TIMESTAMP, PROXY_SESSIONID, GLOBAL_UID, INSTANCE_NUMBER, OS_PROCESS, TRANSACTIONID, SCN, SQL_BIND, SQL_TEXT

DBA_COMMON_AUDIT_TRAIL

Uniform Audit Trails

To use database auditing, you must first set the static `AUDIT_TRAIL` parameter to point to a storage location for audit records. This enables database auditing.

The Oracle database tracks the same fields for standard and fine-grained auditing, enabling you to easily analyze database activities. To accomplish this, both the standard audit trail and the fine-grained audit trail have attributes that complement each other.

The extra information that is collected by standard auditing includes:

- The system change number (SCN), which records every change to the system
- The exact SQL text executed by the user and the bind variables used with the SQL text. These columns appear only if you have specified `AUDIT_TRAIL=DB_EXTENDED` in your initialization parameter file.

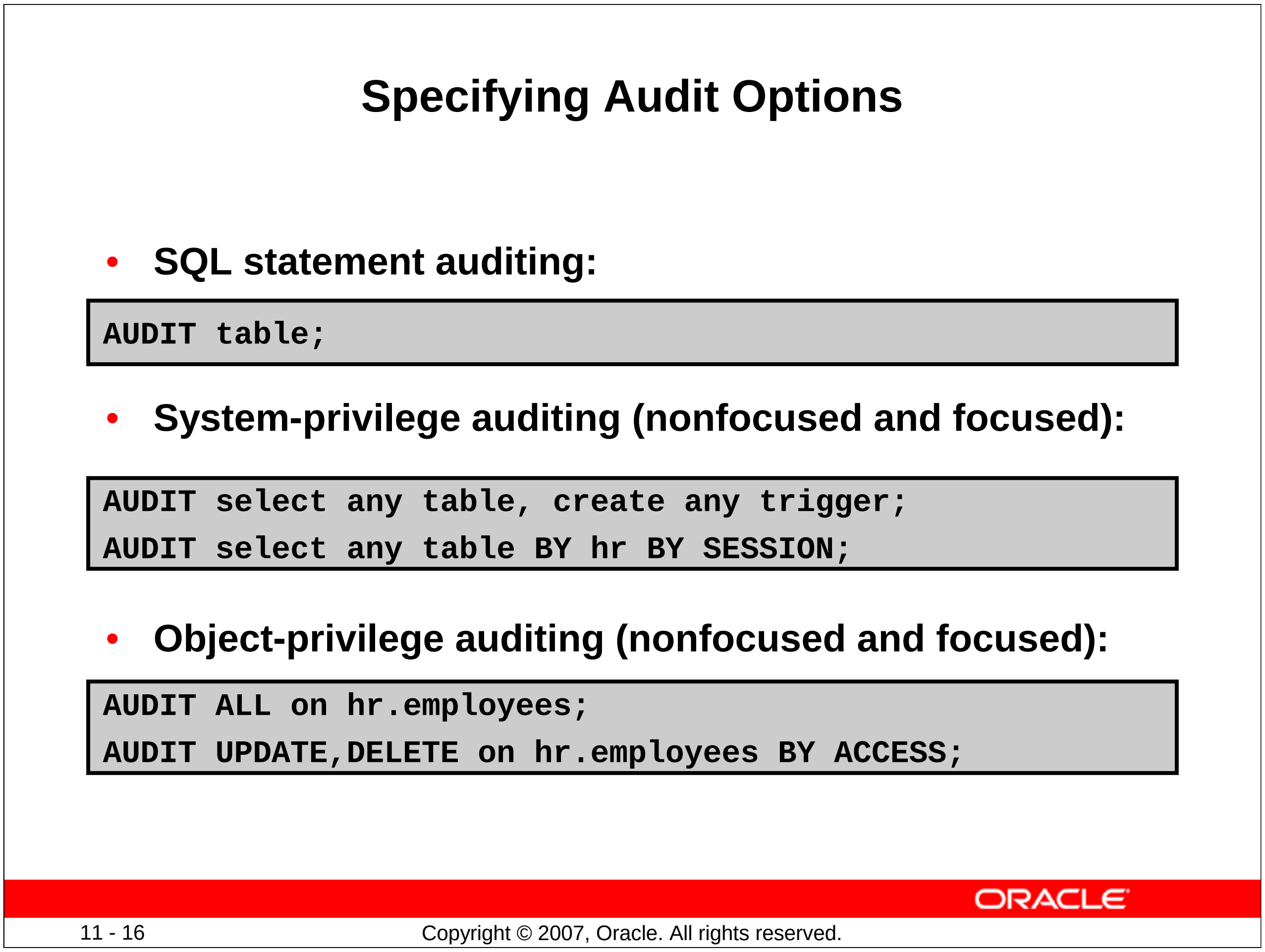
The extra information that is collected by fine-grained auditing includes:

- A serial number for each audit record
- A statement number that links multiple audit entries that originate from a single statement

Common attributes include:

- A global time stamp in Universal Time Coordinates (UTC). This field is useful for monitoring across servers in separate geographic locations and time zones.
- An instance number that is unique for each Real Application Clusters (RAC) instance
- A transaction identifier that helps you group audit records of a single transaction

The `DBA_COMMON_AUDIT_TRAIL` view combines standard and fine-grained audit log records.



Specifying Audit Options

- SQL statement auditing:

```
AUDIT table;
```

- System-privilege auditing (nonfocused and focused):

```
AUDIT select any table, create any trigger;
AUDIT select any table BY hr BY SESSION;
```

- Object-privilege auditing (nonfocused and focused):

```
AUDIT ALL on hr.employees;
AUDIT UPDATE,DELETE on hr.employees BY ACCESS;
```

ORACLE

Specifying Audited Options

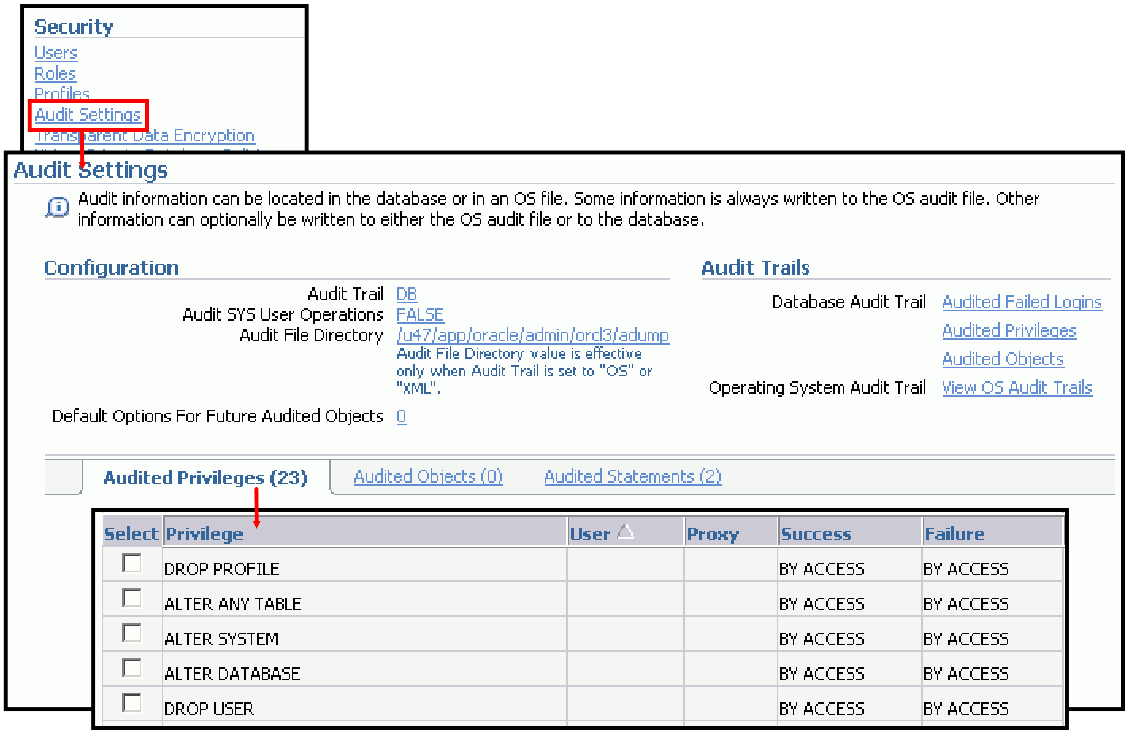
SQL statement auditing: The statement shown in the slide can audit any data definition language (DDL) statement that affects a table, including `CREATE TABLE`, `DROP TABLE`, `TRUNCATE TABLE`, and so on. SQL statement auditing can be focused by username or by success or failure:

```
SQL> AUDIT TABLE BY hr WHENEVER NOT SUCCESSFUL;
```

System-privilege auditing: Can be used to audit the exercise of any system privilege (such as `DROP ANY TABLE`). It can be focused by username or by success or failure. By default, auditing is `BY ACCESS`. Each time an audited system privilege is exercised, an audit record is generated. You can choose to group those records with the `BY SESSION` clause so that only one record is generated per session. (In this way, if a user updates 100,000 records in a table belonging to another user, you gather only one audit record.) Consider using the `BY SESSION` clause to limit the performance and storage impact of system-privilege auditing.

Object-privilege auditing: Can be used to audit actions on tables, views, procedures, sequences, directories, and user-defined data types. This type of auditing can be focused by success or failure and grouped by session or access. Unlike system-privilege auditing, the default grouping is by session. You must explicitly specify `BY ACCESS` if you want a separate audit trail record to be generated for each action.

Enterprise Manager Audit Page



Security
Users
Roles
Profiles
Audit Settings
Transparent Data Encryption

Audit Settings

Audit information can be located in the database or in an OS file. Some information is always written to the OS audit file. Other information can optionally be written to either the OS audit file or to the database.

Configuration

Audit Trail: DB
Audit SYS User Operations: FALSE
Audit File Directory: /u47/app/oracle/admin/orcl3/adump
Audit File Directory value is effective only when Audit Trail is set to "OS" or "XML".
Default Options For Future Audited Objects: 0

Audit Trails

Database Audit Trail: Audited Failed Logins, Audited Privileges, Audited Objects
Operating System Audit Trail: View OS Audit Trails

Audited Privileges (23) | Audited Objects (0) | Audited Statements (2)

Select	Privilege	User △	Proxy	Success	Failure
□	DROP PROFILE			BY ACCESS	BY ACCESS
□	ALTER ANY TABLE			BY ACCESS	BY ACCESS
□	ALTER SYSTEM			BY ACCESS	BY ACCESS
□	ALTER DATABASE			BY ACCESS	BY ACCESS
□	DROP USER			BY ACCESS	BY ACCESS

Enterprise Manager Audit Page

You can reach the Audit page from the Database Control Home page by clicking the Server tab and then clicking the Audit Settings link in the Security region.

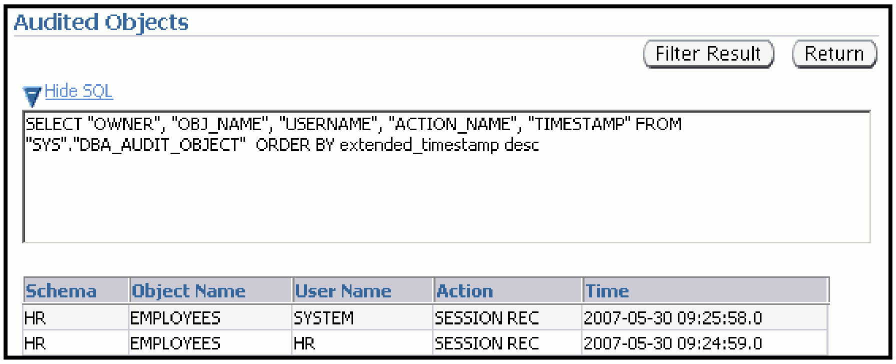
The Audit page contains the following regions:

- **Configuration:** Shows the current configuration parameter values and contains links to edit the parameter values
- **Audit Trails:** Provides easy-to-use access to the audit information that has been collected

Use these tabbed pages to set and unset audit options:

- **Audited Privileges:** Shows privileges that are audited
- **Audited Objects:** Shows objects that are audited
- **Audited Statements:** Shows statements that are audited

Using and Maintaining Audit Information

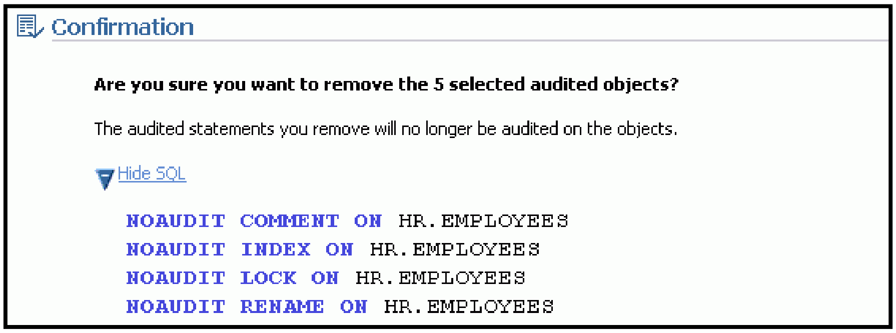


Audited Objects

```
SELECT "OWNER", "OBJ_NAME", "USERNAME", "ACTION_NAME", "TIMESTAMP" FROM
"SYS"."DBA_AUDIT_OBJECT"  ORDER BY extended_timestamp desc
```

Schema	Object Name	User Name	Action	Time
HR	EMPLOYEES	SYSTEM	SESSION REC	2007-05-30 09:25:58.0
HR	EMPLOYEES	HR	SESSION REC	2007-05-30 09:24:59.0

Disable audit options if you are not using them.



Confirmation

Are you sure you want to remove the 5 selected audited objects?

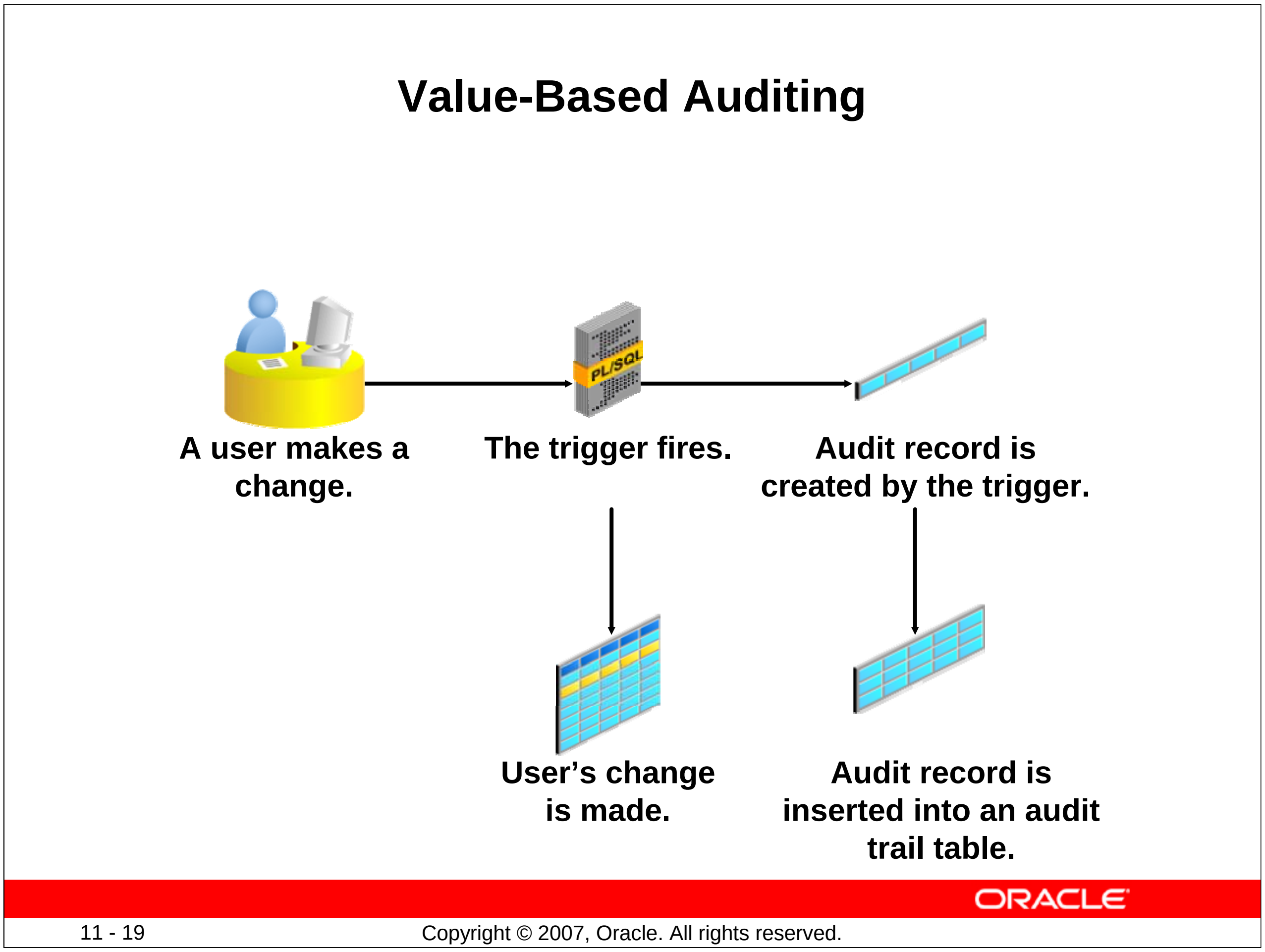
The audited statements you remove will no longer be audited on the objects.

```
NOAUDIT COMMENT ON HR.EMPLOYEES
NOAUDIT INDEX ON HR.EMPLOYEES
NOAUDIT LOCK ON HR.EMPLOYEES
NOAUDIT RENAME ON HR.EMPLOYEES
```

Using and Maintaining Audit Information

Best Practice Tip

Auditing incurs a performance penalty proportional to the number of writes to the audit trail. To tailor the audit options to the needs of your site, enable only those options that are necessary to meet the security policy. Focus the auditing to reduce the number of audit trail entries.



Value-Based Auditing

A user makes a change.

The trigger fires.

Audit record is created by the trigger.

User’s change is made.

Audit record is inserted into an audit trail table.

Value-Based Auditing

Database auditing records the inserts, updates, and deletes that have occurred in audited objects but does not capture the actual values that are changed. To extend database auditing, value-based auditing leverages database triggers (event-driven PL/SQL constructs) to capture the changed values.

When a user inserts, updates, or deletes data from a table with the appropriate trigger attached, the trigger works in the background to copy audit information to a table that is designed to contain the audit information. Value-based auditing tends to degrade performance more than standard database auditing because the audit trigger code must be executed each time the insert, update, or delete operation occurs. The degree of degradation depends on the efficiency of the trigger code. Value-based auditing must be used only in situations in which the information captured by standard database auditing is insufficient.

Value-based auditing is implemented by user or third-party code. The Oracle database provides the PL/SQL constructs to allow value-based audit systems to be built.

Value-Based Auditing (continued)

The key to value-based auditing is the audit trigger, which is simply a PL/SQL trigger that is constructed to capture audit information.

Example of a typical audit trigger:

```
CREATE OR REPLACE TRIGGER system.hrsalary_audit
      AFTER UPDATE OF salary
      ON hr.employees
      REFERENCING NEW AS NEW OLD AS OLD
      FOR EACH ROW
  BEGIN
   IF :old.salary != :new.salary THEN
       INSERT INTO system.audit_employees
       VALUES (sys_context('userenv','os_user'), sysdate,
       sys_context('userenv','ip_address'),
       :new.employee_id ||
         ' salary changed from '||:old.salary||
       ' to '||:new.salary);
  END IF;
  END;
/
```

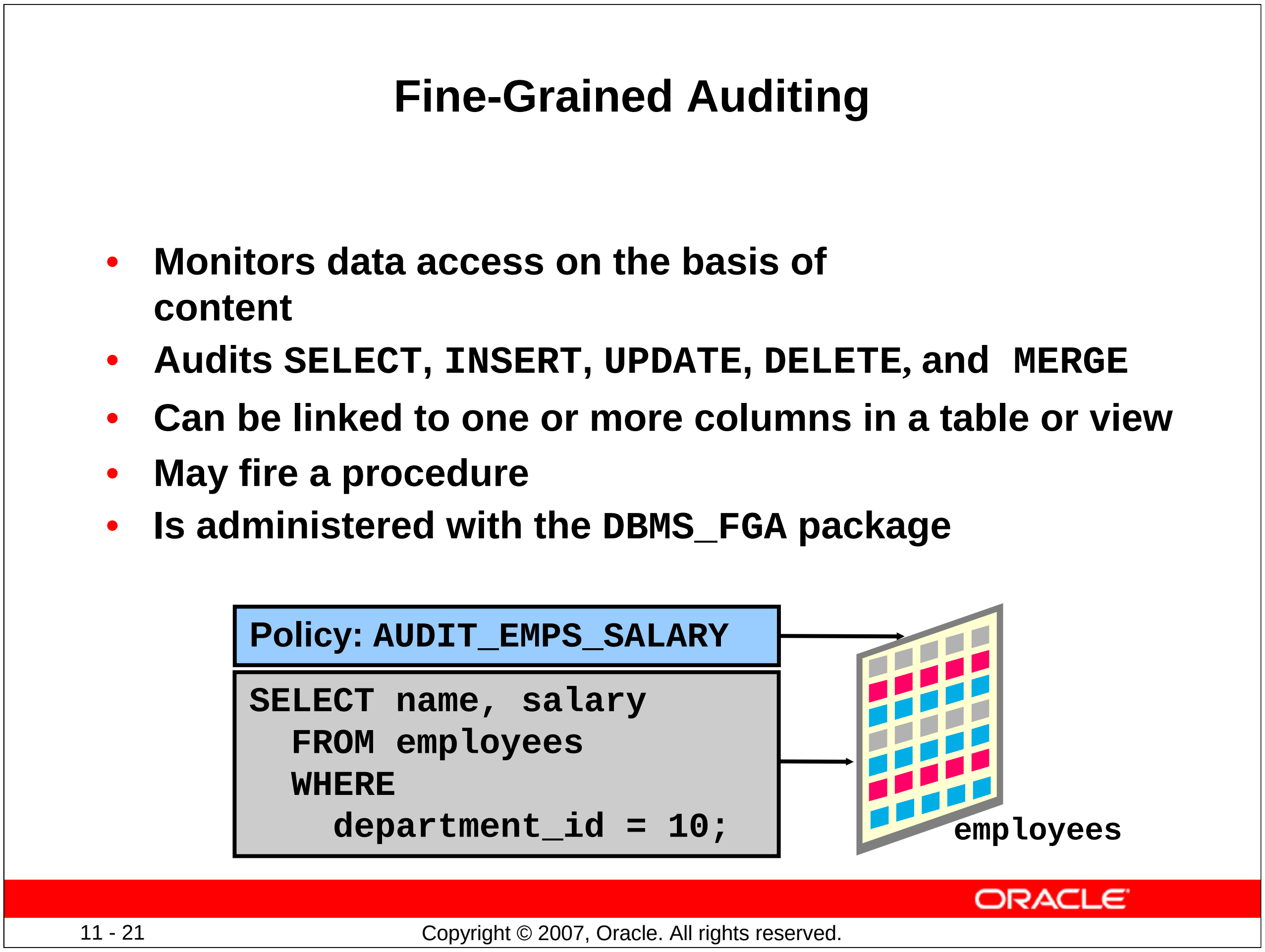
This trigger focuses auditing to capture changes to the salary column of the `hr.employees` table. When a row is updated, the trigger checks the salary column. If the old salary is not equal to the new salary, the trigger inserts an audit record into the `audit_employees` table (created via a separate operation in the `SYSTEM` schema). The audit record includes the username, the IP address from which the change is made, the primary key identifying which record is changed, and the actual salary values that are changed.

Database triggers can also be used to capture information about user connections in cases where standard database auditing does not gather sufficient data. With login triggers, the administrator can capture data that identifies the user who is connecting to the database. Examples include the following:

- IP address of the person logging in
- First 48 characters of the program name that is used to connect to the instance
- Terminal name that is used to connect to the instance

For a complete list of user parameters, see the section titled “`SYS_CONTEXT`” in the *Oracle Database SQL Reference*.

Value-based triggers have been superceded in many cases by the fine-grained auditing (FGA) feature.



Fine-Grained Auditing

- Monitors data access on the basis of content
- Audits `SELECT`, `INSERT`, `UPDATE`, `DELETE`, and `MERGE`
- Can be linked to one or more columns in a table or view
- May fire a procedure
- Is administered with the `DBMS_FGA` package

Policy: AUDIT_EMPS_SALARY

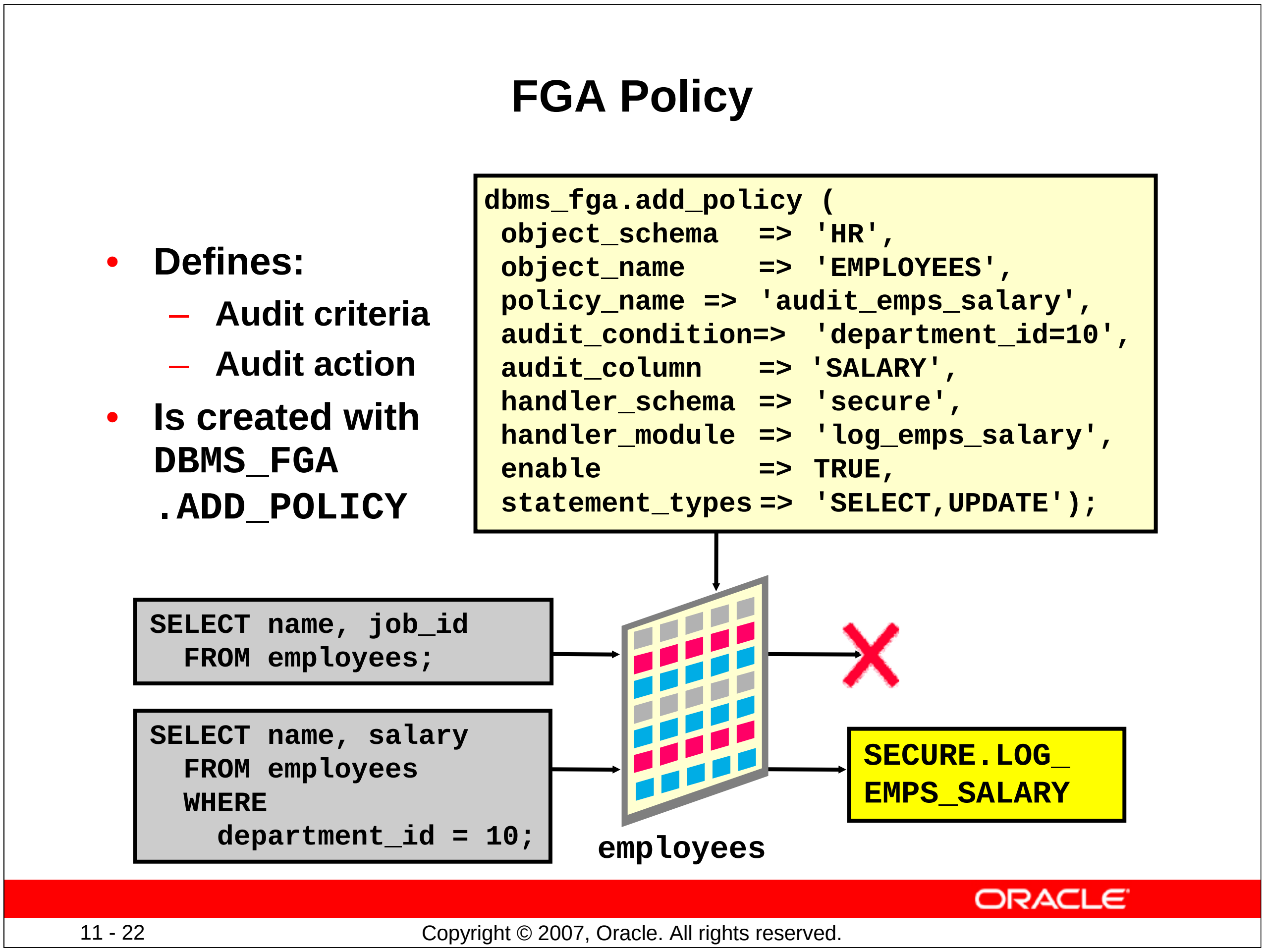
```
SELECT name, salary
  FROM employees
  WHERE
    department_id = 10;
```

Fine-Grained Auditing

Database auditing records the fact that an operation has occurred but does not capture information about the statement that caused the operation. Fine-grained auditing (FGA) extends that capability to enable the capture of actual SQL statements that query or manipulate data. FGA also allows auditing to be more narrowly focused than standard or value-based database auditing.

FGA options can be focused by individual columns in a table or view, and can even be conditional so that audits are captured only if certain administrator-defined specifications are met. More than one relevant column is supported for an FGA policy. By default, if any one of these columns is present in the SQL statement, it is audited. `DBMS_FGA.ALL_COLUMNS` and `DBMS_FGA.ANY_COLUMNS` are provided to audit on the basis of whether any or all of the relevant columns are used in the statement.

Use the `DBMS_FGA` PL/SQL package to create an audit policy on the target table or view. If any of the rows returned from a query block match the audited column and the specified audit condition, an audit event causes an audit record to be created and stored in the audit trail. As an option, the audit event can also execute a procedure. FGA automatically focuses auditing at the statement level. A `SELECT` statement that returns thousands of rows thus generates only one audit record.



FGA Policy

- Defines:
  - Audit criteria
  - Audit action
- Is created with DBMS_FGA .ADD_POLICY

```
dbms_fga.add_policy (
 object_schema   => 'HR',
 object_name     => 'EMPLOYEES',
 policy_name => 'audit_emps_salary',
 audit_condition=> 'department_id=10',
 audit_column    => 'SALARY',
 handler_schema  => 'secure',
 handler_module  => 'log_emps_salary',
 enable          => TRUE,
 statement_types => 'SELECT,UPDATE');
```

```
SELECT name, job_id
  FROM employees;
```

```
SELECT name, salary
  FROM employees
  WHERE
    department_id = 10;
```

FGA Policy

The example in the slide shows the creation of a fine-grained auditing policy with the DBMS_FGA.ADD_POLICY procedure, which accepts the following arguments.

Policy Name

You assign each FGA policy a name when you create it. The example in the slide names the policy AUDIT_EMPS_SALARY by using the following argument:

```
policy_name => 'audit_emps_salary'
```

Audit Condition

The audit condition is a SQL predicate that defines when the audit event must fire. In the slide example, all rows in department 10 are audited by using the following condition argument:

```
audit_condition => 'department_id = 10'
```

FGA Policy (continued)

Audit Column

The audit column defines the data that is being audited. An audit event occurs if this column is included in the `SELECT` statement or if the audit condition allows the selection. The example in the slide audits two columns by using the following argument:

```
        audit_column => 'SALARY,COMMISION_PCT'
```

This argument is optional. If it is not specified, only the `AUDIT_CONDITION` argument determines whether an audit event must occur.

Object

The object is the table or view that is being audited. It is passed as two arguments:

- The schema that contains the object
- The name of the object

The example in the slide audits the `hr.employees` table by using the following arguments:

```
        object_schema => 'hr'
        object_name => 'employees'
```

Handler

An optional event handler is a PL/SQL procedure that defines additional actions that must be taken during auditing. For example, the event handler can send an alert page to the administrator. If it is not defined, an audit event entry is inserted into the audit trail. If an audit event handler is defined, the audit entry is inserted into the audit trail and the audit event handler is executed.

The audit event entry includes the FGA policy that caused the event, the user executing the SQL statement, and the SQL statement and its bind variables.

The event handler is passed as two arguments:

- The schema that contains the PL/SQL program unit
- The name of the PL/SQL program unit

The example in the slide executes the `SECURE.LOG_EMPS_SALARY` procedure by using the following arguments:

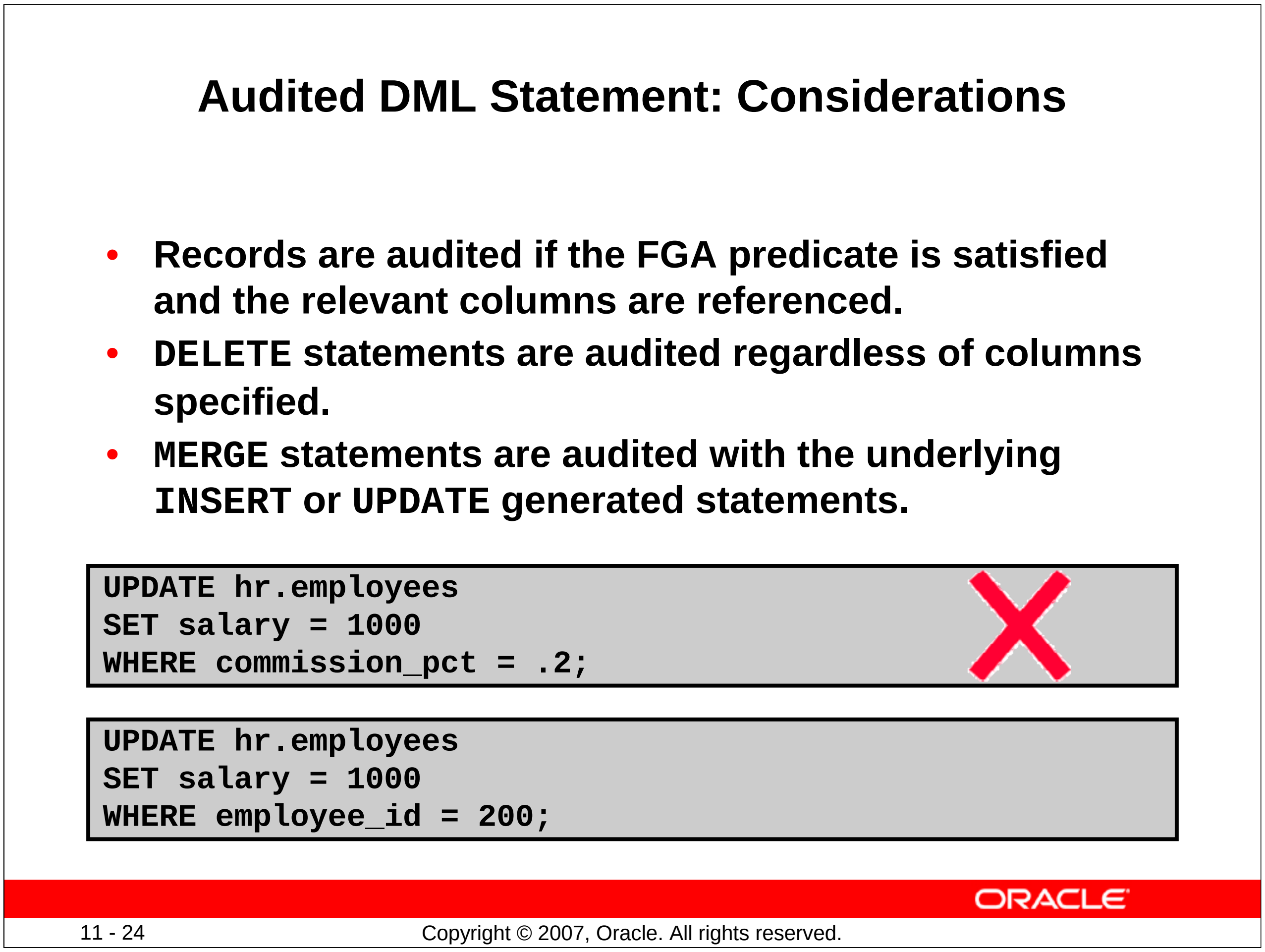
```
        handler_schema => 'secure'
        handler_module => 'log_emps_salary'
```

By default, audit trail always writes the SQL text and SQL bind information to LOBs. The default can be changed (for example, if the system would suffer performance degradation).

Status

The status indicates whether the FGA policy is enabled. In the slide example, the following argument enables the policy:

```
        enable => TRUE
```



Audited DML Statement: Considerations

- Records are audited if the FGA predicate is satisfied and the relevant columns are referenced.
- `DELETE` statements are audited regardless of columns specified.
- `MERGE` statements are audited with the underlying `INSERT` or `UPDATE` generated statements.

```
UPDATE hr.employees
SET salary = 1000
WHERE commission_pct = .2;
```

✗

```
UPDATE hr.employees
SET salary = 1000
WHERE employee_id = 200;
```

11 - 24

Audited DML Statement: Considerations

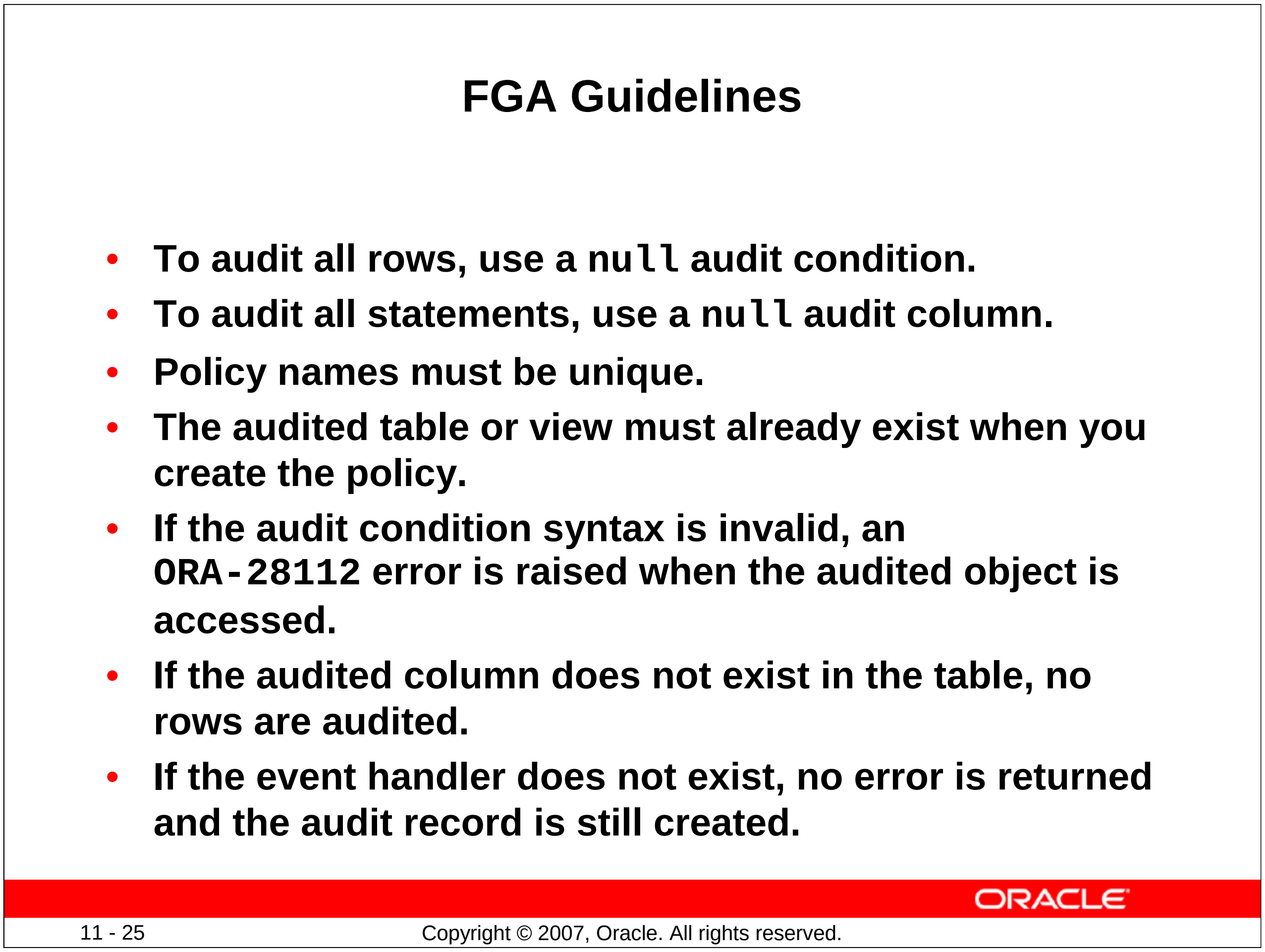
With an FGA policy defined for DML statements, a DML statement is audited if the data rows (both new and old) that are being manipulated meet the policy predicate criteria.

However, if relevant columns are also specified in the policy definition, the statement is audited when the data meets the FGA policy predicate and the statement references the relevant columns defined.

For `DELETE` statements, specifying relevant columns during policy definition is not useful because all columns in a table are touched by a `DELETE` statement. Therefore, a `DELETE` statement is always audited regardless of the relevant columns.

`MERGE` statements are supported by FGA. The underlying `INSERT` and `UPDATE` statements are audited if they meet the defined `INSERT` or `UPDATE` FGA policies.

Using the previously defined FGA policy, the first statement is not audited whereas the second one is. None of the employees in department 10 receive a commission, but `employee_id=200` specifies an employee in department 10.



FGA Guidelines

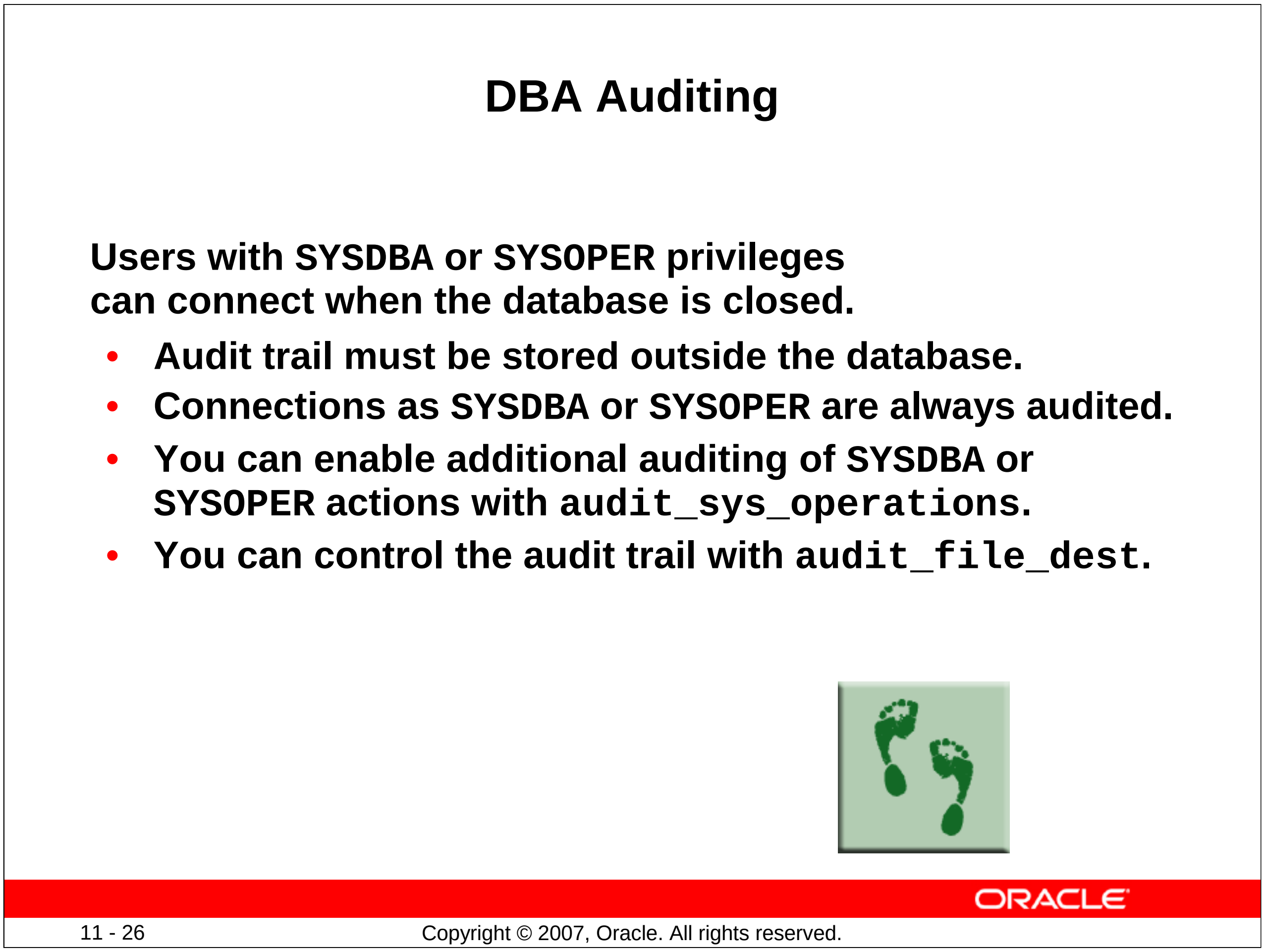
- To audit all rows, use a `null` audit condition.
- To audit all statements, use a `null` audit column.
- Policy names must be unique.
- The audited table or view must already exist when you create the policy.
- If the audit condition syntax is invalid, an `ORA-28112` error is raised when the audited object is accessed.
- If the audited column does not exist in the table, no rows are audited.
- If the event handler does not exist, no error is returned and the audit record is still created.

FGA Guidelines

For the `SELECT` statements, FGA captures the statement itself and not the actual rows. However, when FGA is combined with Flashback Query, the rows can be reconstructed as they existed at that point in time.

For more details about Flashback Query, see the lesson titled “Performing Flashback.”

For more details about the `DBMS_FGA` package, see the *Oracle Database PL/SQL Packages and Types Reference*.



DBA Auditing

Users with SYSDBA or SYSOPER privileges can connect when the database is closed.

- **Audit trail must be stored outside the database.**
- **Connections as SYSDBA or SYSOPER are always audited.**
- **You can enable additional auditing of SYSDBA or SYSOPER actions with `audit_sys_operations`.**
- **You can control the audit trail with `audit_file_dest`.**

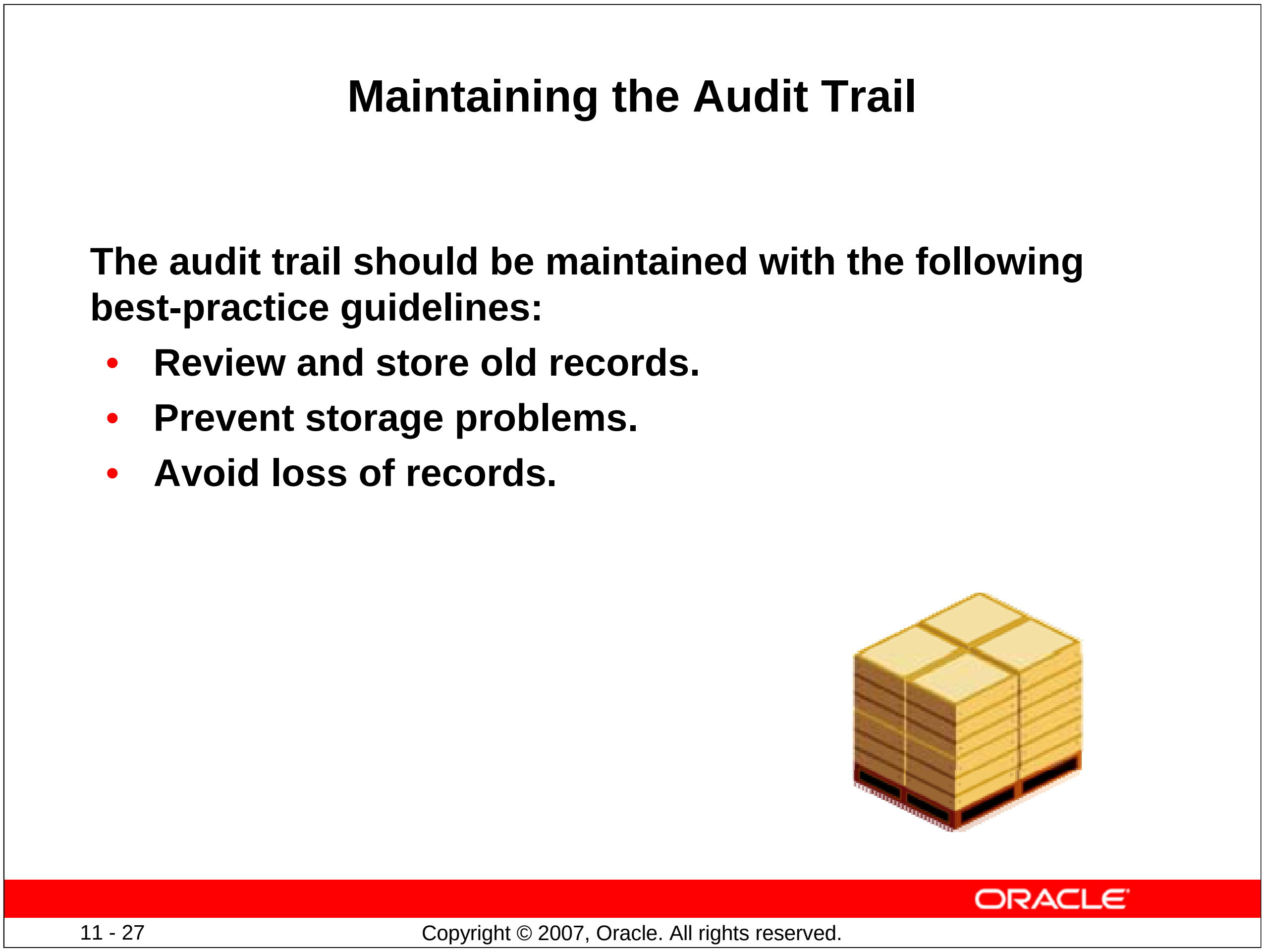
DBA Auditing

The SYSDBA and SYSOPER users have privileges to start up and shut down the database. Because they may make changes while the database is closed, the audit trail for these privileges must be stored outside the database. The Oracle database automatically captures login events by the SYSDBA and SYSOPER users. This provides a valuable way to track authorized or unauthorized SYSDBA and SYSOPER actions, but it is useful only if the OS audit trail is reviewed.

The Oracle database always captures the login events of privileged users. Other actions are captured if DBA auditing is specifically enabled. Enable auditing of the SYSDBA and SYSOPER users by setting the initialization parameter:

`audit_sys_operations=TRUE` (The default is FALSE.)

If the SYS operations are audited, the `audit_file_dest` initialization parameter controls the storage location of the audit records. On a Windows platform, the audit trail defaults to the Windows event log. On UNIX and Linux platforms, audit records are stored in `$ORACLE_HOME/rdbms/audit`.



Maintaining the Audit Trail

The audit trail should be maintained with the following best-practice guidelines:

- **Review and store old records.**
- **Prevent storage problems.**
- **Avoid loss of records.**

11 - 27

Maintaining the Audit Trail

Each type of audit trail must be maintained. Basic maintenance must include reviewing the audit records and removing older records from the database or operating system. Audit trails can grow to fill the available storage. If the file system is full, the system may crash or simply cause performance problems. If the database audit trail fills the tablespace, audited actions do not complete. If the audit trail fills the system tablespace, the performance of other operations is affected before audit operations halt.

The audit trail for standard auditing is stored in the `AUD$` table. The audit trail for FGA is the `FGA_LOG$` table. Both these tables are created in the `SYSTEM` tablespace by default. You can move these tables to another tablespace by using the Data Pump export and import utilities.

Note: Moving the audit tables out of the `SYSTEM` tablespace is not supported.

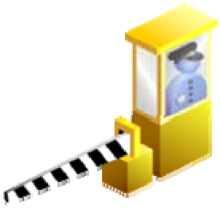
Audit records can be lost during the process of removing records from the audit tables.

Best Practice Tip

Use an export based on a time stamp, and then delete rows from the audit trail based on the same time stamp.

Security Updates

- Oracle posts Critical Patch Update information on the Oracle Technology Network (OTN) site at: `http://www.oracle.com/technology/deploy/security/alerts.htm`
- Oracle database administrators and developers can also subscribe to email notification of security updates from this OTN page.



Security Updates

Oracle Critical Patch Updates (CPUs) contain an assessment of the risk and degree of exposure associated with the vulnerability, along with a severity rating. The CPUs help to mitigate this risk; you can manage patches with Oracle Enterprise Manager. Oracle Corporation includes in the CPU an acknowledgement of the individual or organization that notified it of the vulnerability.

CPU information is posted on the Oracle Technology Network site and on Oracle MetaLink. Although CPU information is publicly available for anyone interested in the updates, only customers with a current Customer Support Identification (CSI) number can download a patch.

Oracle appreciates your cooperation in keeping its products secure through prompt, complete, and confidential notification of potential security vulnerabilities. If you discover a security vulnerability with any Oracle product, we encourage you to submit a service request through MetaLink. You can also send email to `secalert_us@oracle.com`.

Applying Security Patches

- Use the Critical Patch Update process.
- Apply all security patches and workarounds.
- Contact the Oracle Security product team.

Applying Security Patches

Critical Patch Update Process

Oracle initiated the CPU process in January 2005. The process bundles together critical patches on a quarterly basis. (This program replaced the Security Alert patch releases.) These patches are cumulative and include commonly requested and required prerequisite patches. The quarterly patch release comes with a risk assessment matrix to enable you to determine for your site the impact and security risks. (See MetaLink Note: 360470.1 “Security Alerts and Critical Patch Updates–Frequently Asked Questions.”) You must subscribe to MetaLink to receive Critical Patch Updates.

Apply All Security Patches and Workarounds

Always apply all relevant and current security patches for both the operating system on which the database resides and the Oracle software, and for all installed options and components.

Contact the Oracle Security Products Team

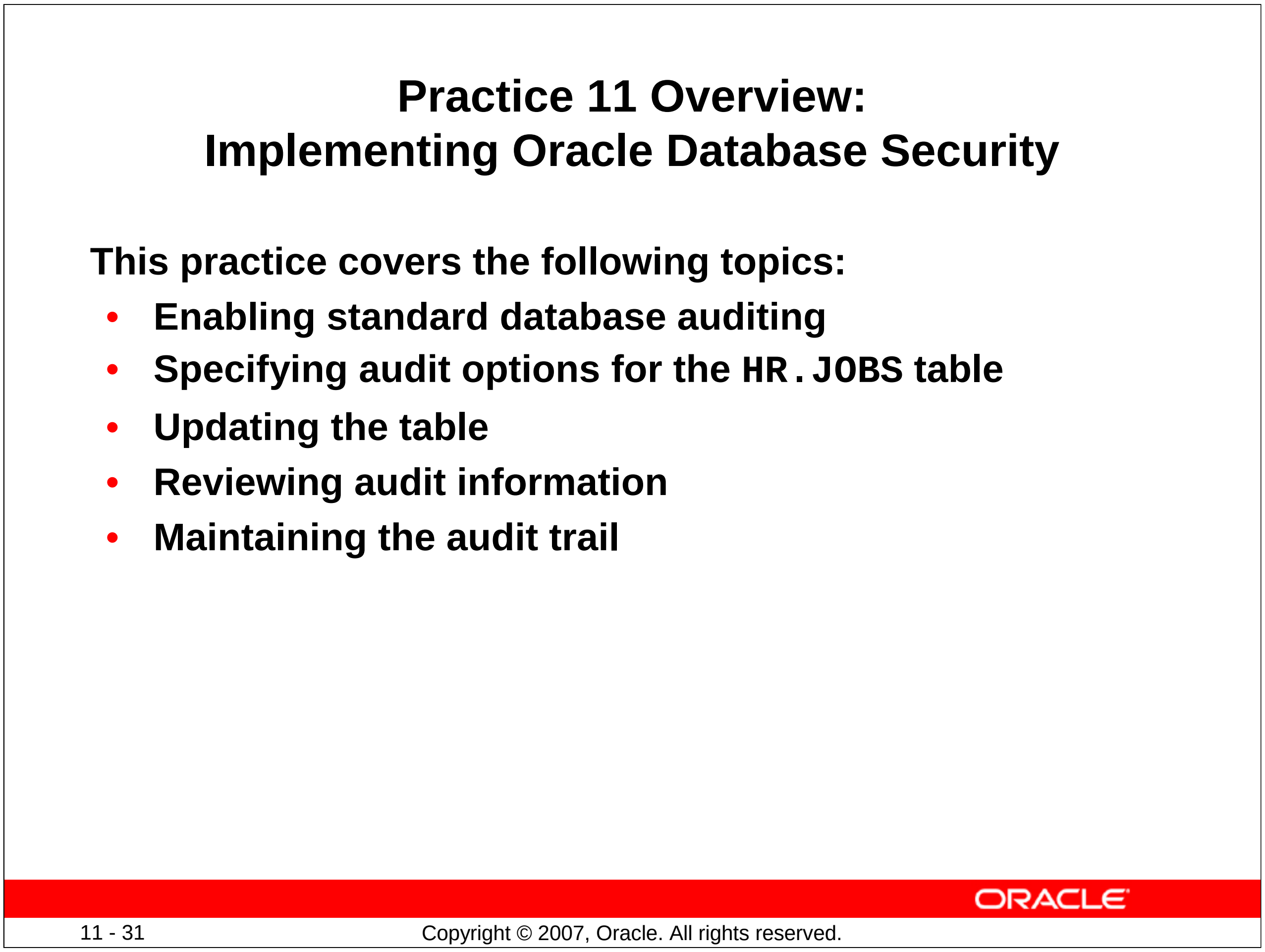
If you believe that you have found a security vulnerability in Oracle software, follow the instructions provided from the Reporting Security Vulnerabilities link at this URL:

http://www.oracle.com/technology/deploy/security/alerts.htm

Summary

In this lesson, you should have learned how to:

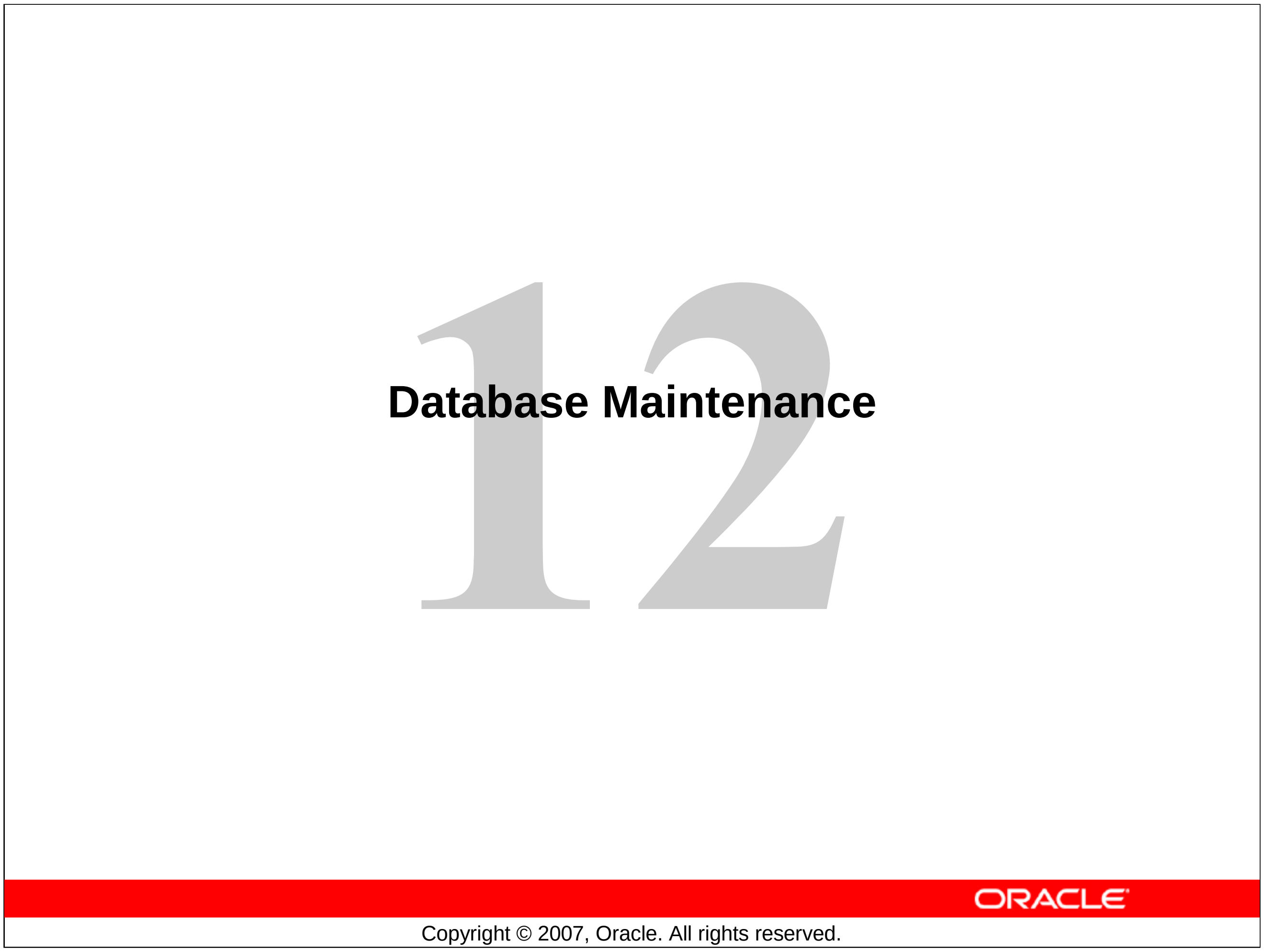
- Describe DBA responsibilities for security
- Apply the principle of least privilege
- Enable standard database auditing
- Specify audit options
- Review audit information
- Maintain the audit trail



Practice 11 Overview: Implementing Oracle Database Security

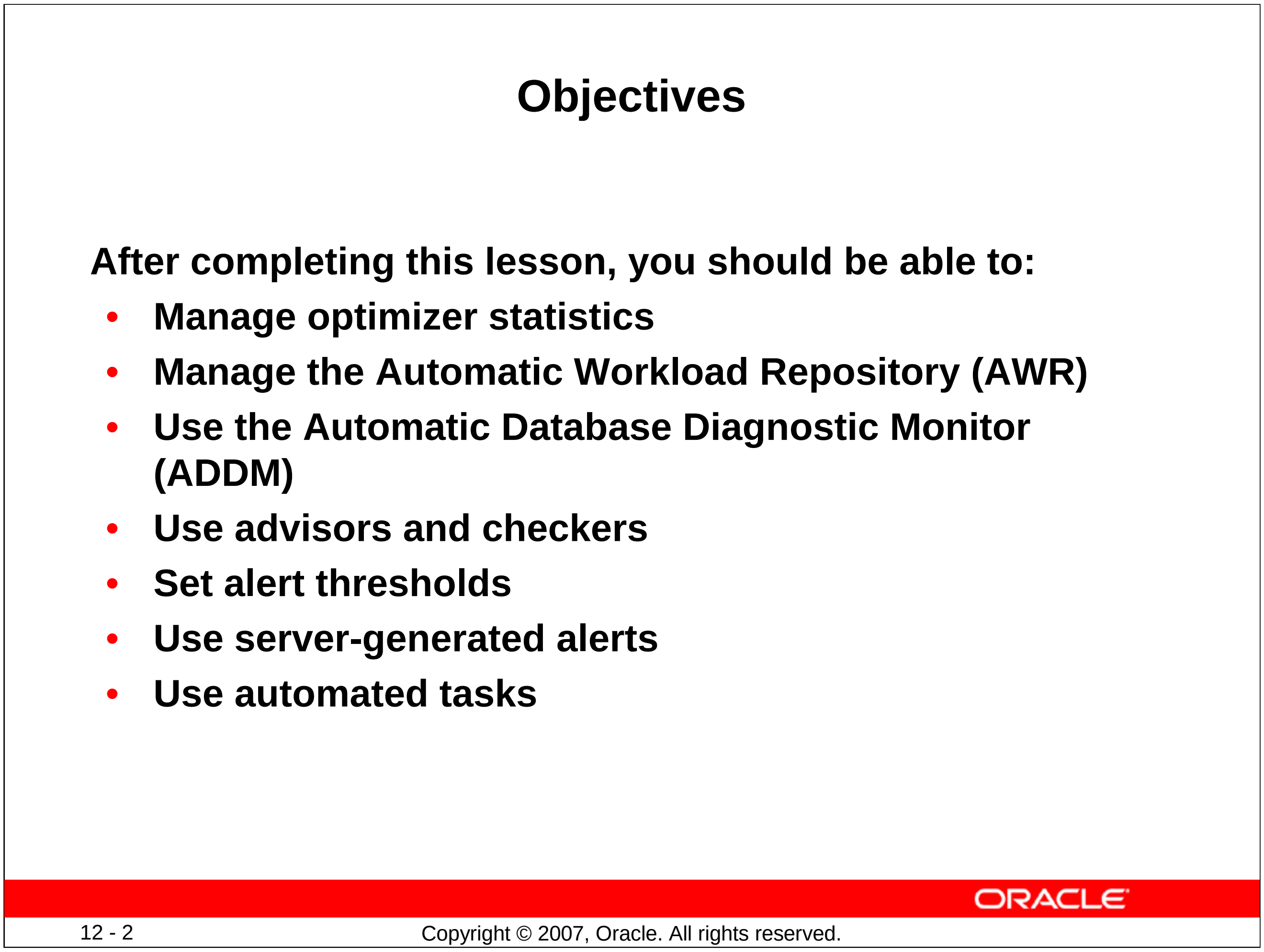
This practice covers the following topics:

- Enabling standard database auditing
- Specifying audit options for the `HR.JOBS` table
- Updating the table
- Reviewing audit information
- Maintaining the audit trail



12

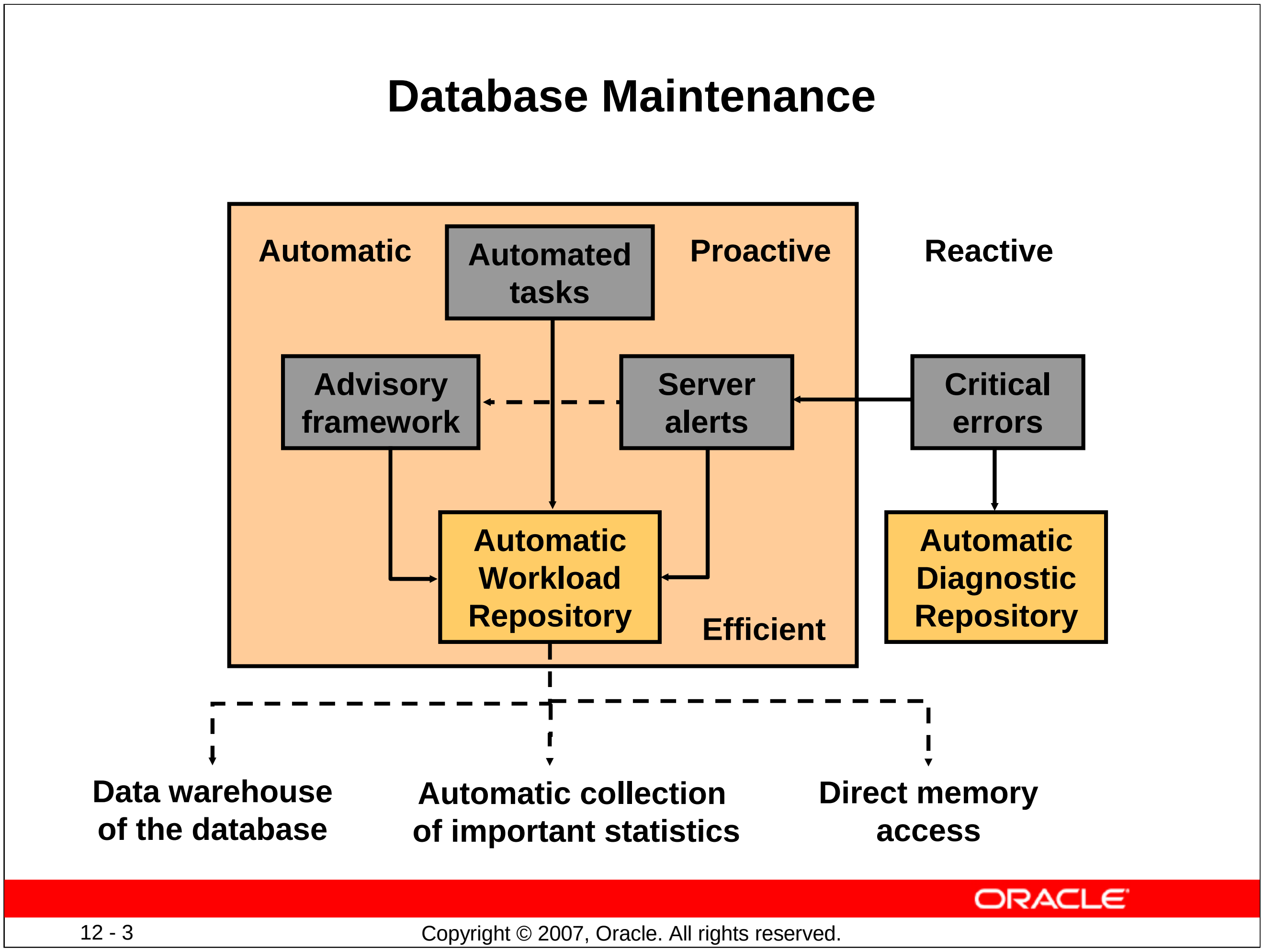
Database Maintenance



Objectives

After completing this lesson, you should be able to:

- Manage optimizer statistics
- Manage the Automatic Workload Repository (AWR)
- Use the Automatic Database Diagnostic Monitor (ADDM)
- Use advisors and checkers
- Set alert thresholds
- Use server-generated alerts
- Use automated tasks



Database Maintenance

Automatic

Automated tasks

Proactive

Reactive

Advisory framework

Server alerts

Critical errors

Automatic Workload Repository

Efficient

Automatic Diagnostic Repository

Data warehouse of the database

Automatic collection of important statistics

Direct memory access

Database Maintenance

Proactive database maintenance is made easy by the sophisticated infrastructure of the Oracle database, including the following main elements:

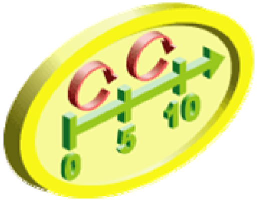
- The Automatic Workload Repository (AWR) is a built-in repository in each Oracle database. At regular intervals, the Oracle database makes a snapshot of all its vital statistics and workload information and stores this data in the AWR. The captured data can be analyzed by you, by the database itself, or by both.
- Using automated tasks, the database performs routine maintenance operations such as regular backups, refreshing optimizer statistics, and database health checks.

Reactive database maintenance includes critical errors and conditions discovered by database health checkers:

- For problems that cannot be resolved automatically and require administrators to be notified (such as running out of space), the Oracle database provides server-generated alerts. The Oracle database by default monitors itself and sends out alerts to notify you of problems. The alerts notify you and often also provide recommendations on how to resolve reported problem.
- Recommendations are generated from a number of advisors, each of which is responsible for a subsystem. For example, there are memory, segment, and SQL advisors.

Terminology

- Automatic Workload Repository (AWR): Infrastructure for data gathering, analysis, and solutions recommendations
- Baseline: A set of AWR snapshots for performance comparison
- Metric: Rate of change in a cumulative statistic
- Statistics: Data collections used for performance monitoring or SQL optimization
- Threshold: A boundary value against which metric values are compared



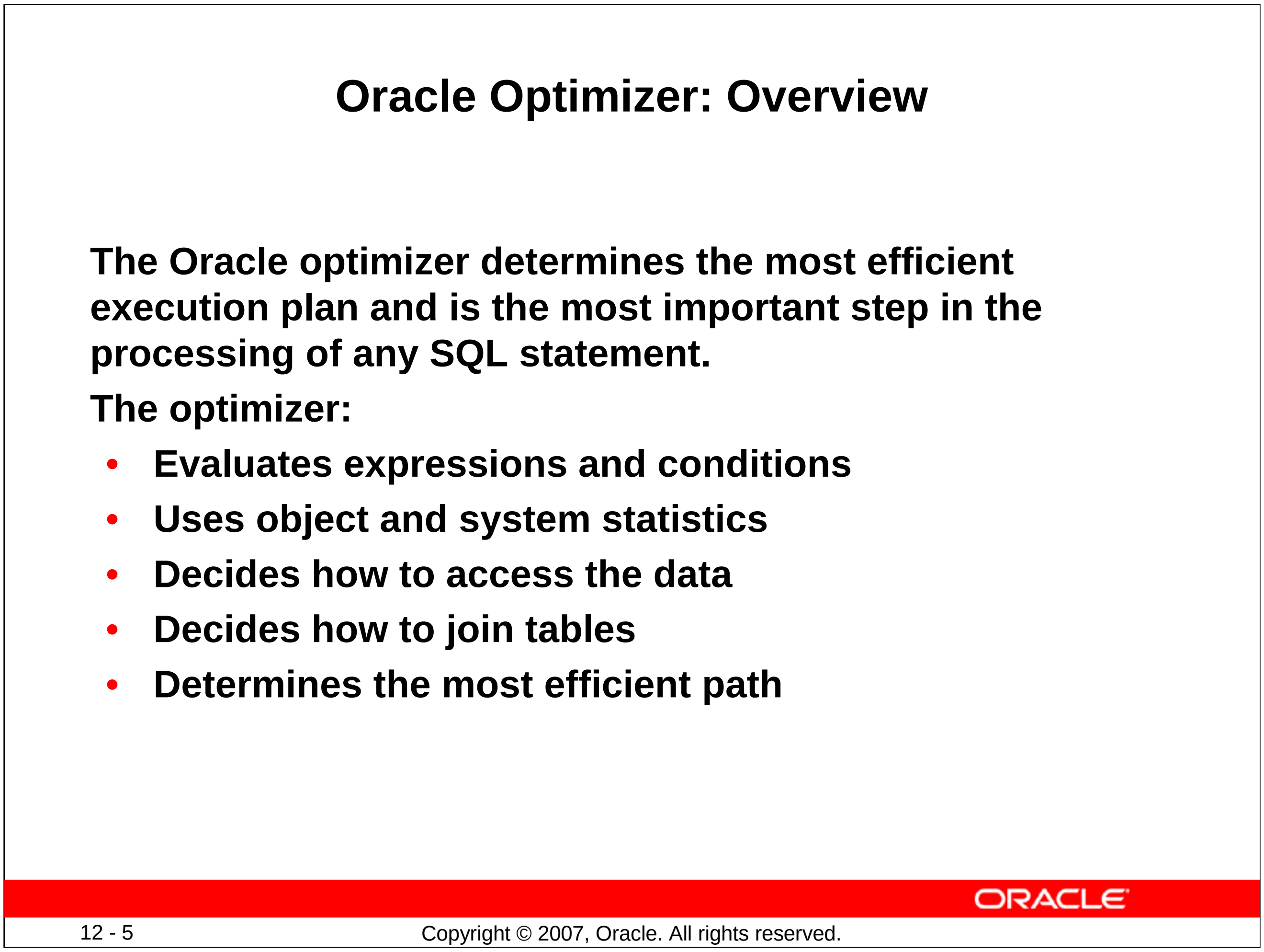
Terminology

The *Automatic Workload Repository* (AWR) provides services to internal Oracle server components to collect, process, maintain, and use performance statistics for problem detection and self-tuning purposes. *Active Session History* (ASH) is the history of recent session activity stored in the AWR.

Statistics are collections of data that provide more details about the database and the objects in it. Optimizer statistics are used by the query optimizer to choose the best execution plan for each SQL statement. Database statistics provide information for performance monitoring.

AWR snapshots include database statistics and metrics, application statistics (transaction volumes, response time), operating system statistics, and other measures. A *baseline* is a set of AWR snapshots collected over a period of time. The baseline is used for performance comparison, either current performance versus the baseline or one baseline compared to another.

The *System Moving Window* baseline is collected by default in Oracle Database 11*g*. The System Moving Window baseline is a changing set of snapshots that include the last eight days of snapshots by default. This baseline becomes valid after sufficient data has been collected and the statistics calculation occurs. The statistics calculation is scheduled for every Saturday at midnight by default.



Oracle Optimizer: Overview

The Oracle optimizer determines the most efficient execution plan and is the most important step in the processing of any SQL statement.

The optimizer:

- **Evaluates expressions and conditions**
- **Uses object and system statistics**
- **Decides how to access the data**
- **Decides how to join tables**
- **Determines the most efficient path**

Oracle Optimizer: Overview

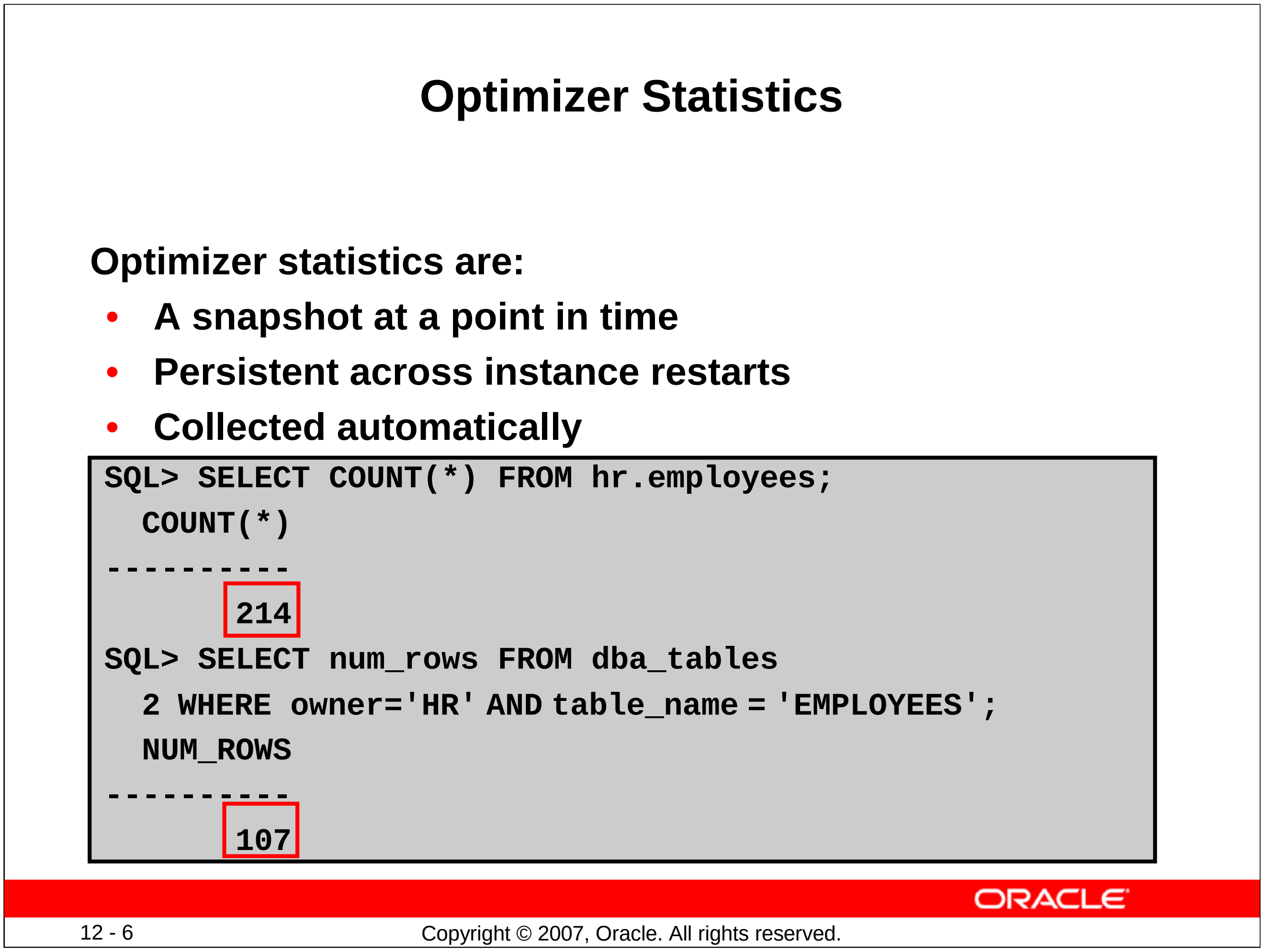
The optimizer is the part of the Oracle database that creates the execution plan for a SQL statement. The determination of the execution plan is an important step in the processing of any SQL statement and can greatly affect execution time.

The execution plan is a series of operations that are performed in sequence to execute the statement. The optimizer considers many factors related to the referenced objects and the conditions specified in the query. The information necessary to the optimizer includes:

- Statistics gathered for the system (I/O, CPU, and so on) as well as schema objects (number of rows, index, and so on)
- Information in the dictionary
- `WHERE` clause qualifiers
- Hints supplied by the developer

When you use diagnostic tools such as Enterprise Manager, `EXPLAIN PLAN`, and SQL*Plus `AUTOTRACE`, you can see the execution plan that the optimizer chooses.

Note: The Oracle optimizer has two names based on its functionality: the *query optimizer* and the *Automatic Tuning Optimizer*.



Optimizer Statistics

Optimizer statistics are:

- A snapshot at a point in time
- Persistent across instance restarts
- Collected automatically

```
SQL> SELECT COUNT(*) FROM hr.employees;
  COUNT(*)
----------
       214
SQL> SELECT num_rows FROM dba_tables
  2 WHERE owner='HR' AND table_name = 'EMPLOYEES';
  NUM_ROWS
----------
       107
```

Optimizer Statistics

Optimizer statistics include table, column, index, and system statistics. Statistics for tables and indexes are stored in the data dictionary. These statistics are not intended to provide real-time data. They provide the optimizer a *statistically* correct snapshot of data storage and distribution, which the optimizer uses to make decisions on how to access data.

The statistics that are collected include:

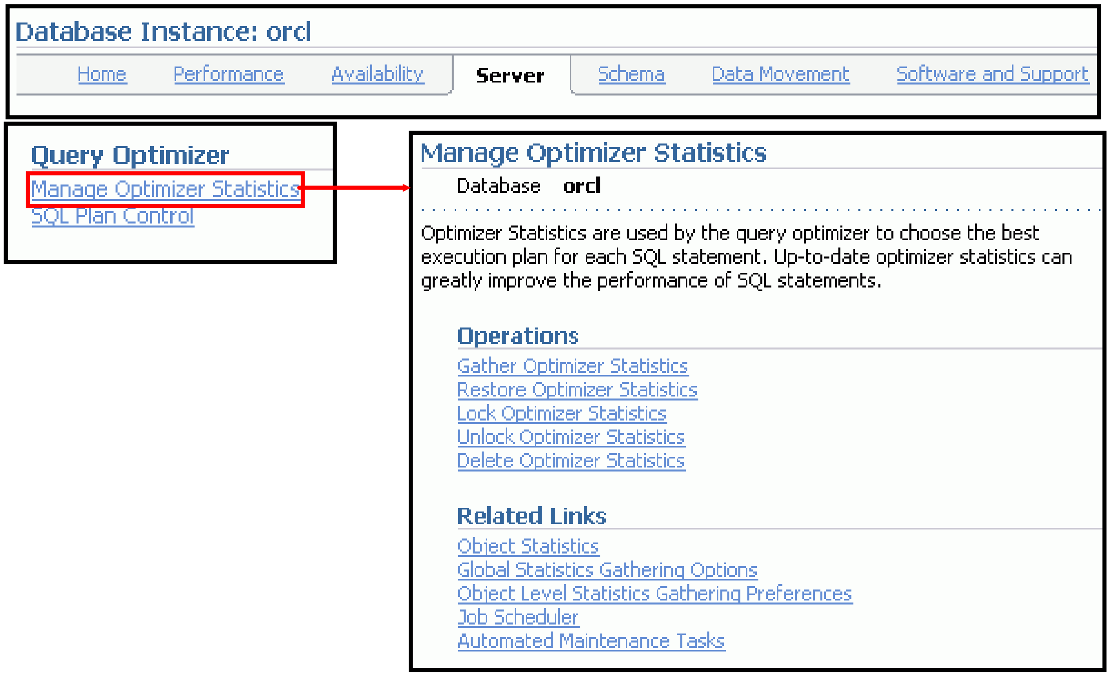
- Size of the table or index in database blocks
- Number of rows
- Average row size and chain count (tables only)
- Height and number of deleted leaf rows (indexes only)

As data is inserted, deleted, and modified, these facts change. Because the performance impact of maintaining real-time data distribution statistics is prohibitive, these statistics are updated by periodically gathering statistics on tables and indexes.

Optimizer statistics are collected automatically by an automatic maintenance job that runs during predefined maintenance windows once daily by default. System statistics are operating system characteristics that are used by the optimizer. These statistics are not collected automatically. For details about collecting system statistics, see the *Oracle Database Performance Tuning Guide*.

Optimizer statistics are not the same as the database performance statistics that are gathered in the AWR snapshot.

Using the Manage Optimizer Statistics Page


Database Instance: orcl
Home
Performance
Availability
Server
Schema
Data Movement
Software and Support
Query Optimizer
Manage Optimizer Statistics
SQL Plan Control

Manage Optimizer Statistics

Database orcl

Optimizer Statistics are used by the query optimizer to choose the best execution plan for each SQL statement. Up-to-date optimizer statistics can greatly improve the performance of SQL statements.

Operations
Gather Optimizer Statistics
Restore Optimizer Statistics
Lock Optimizer Statistics
Unlock Optimizer Statistics
Delete Optimizer Statistics
Related Links
Object Statistics
Global Statistics Gathering Options
Object Level Statistics Gathering Preferences
Job Scheduler
Automated Maintenance Tasks

Using the Manage Optimizer Statistics Page

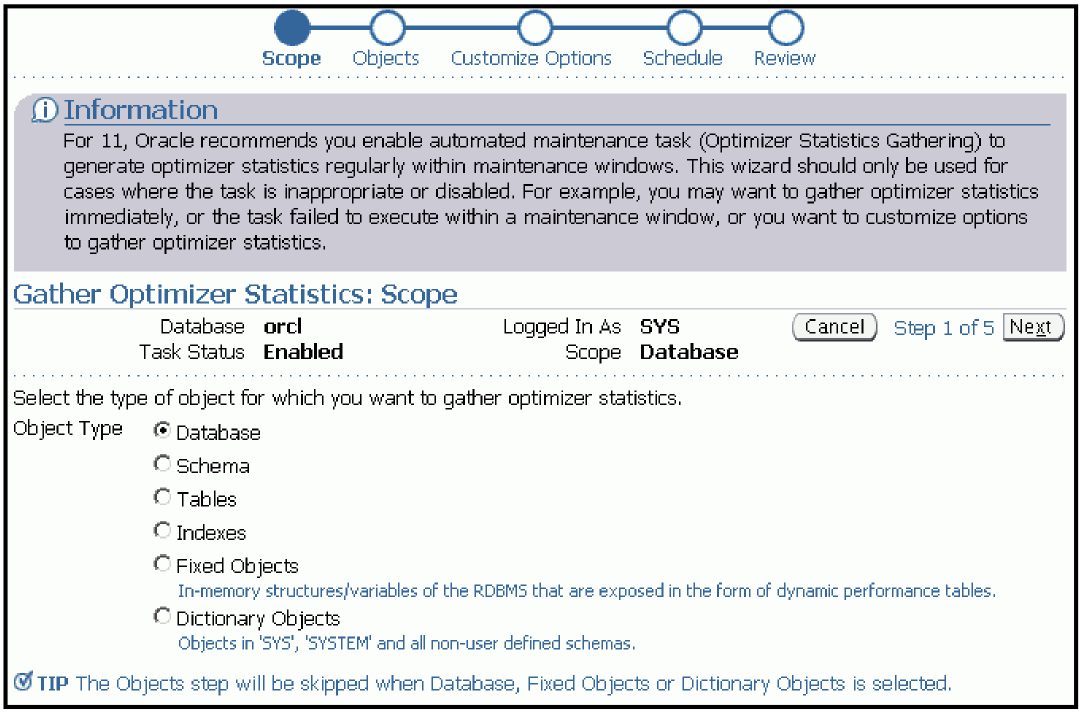
To manage optimizer statistics in Enterprise Manager, click the Server tab and then click Manage Optimizer Statistics under the Query Optimizer section. From this page, you can perform the following tasks on statistics:

- Gather optimizer statistics manually.
- Restore optimizer statistics to a point in the past. The chosen point in time must be within the optimizer statistics retention period, which defaults to 30 days.
- Lock optimizer statistics to guarantee that the statistics for certain objects are never overwritten. This is useful if statistics have been calculated for a certain table at a time when well-representative data is present, and you want to always have those statistics. No fluctuations in the table affect the statistics if they are locked.
- Unlock optimizer statistics to undo the previously done lock.
- Delete optimizer statistics to delete statistics.

Best Practice Tip

Use the automatic maintenance tasks to gather optimizer statistics. To enable the task for gathering optimizer statistics gathering, you must ensure that the `STATISTICS_LEVEL` initialization parameter is set to `TYPICAL` or `ALL`.

Gathering Optimizer Statistics Manually



Information

For 11, Oracle recommends you enable automated maintenance task (Optimizer Statistics Gathering) to generate optimizer statistics regularly within maintenance windows. This wizard should only be used for cases where the task is inappropriate or disabled. For example, you may want to gather optimizer statistics immediately, or the task failed to execute within a maintenance window, or you want to customize options to gather optimizer statistics.

Gather Optimizer Statistics: Scope

Database	orcl	Logged In As	SYS
Task Status	Enabled	Scope	Database

Select the type of object for which you want to gather optimizer statistics.

Object Type
● Database
○ Schema
○ Tables
○ Indexes
○ Fixed Objects
In-memory structures/variables of the RDBMS that are exposed in the form of dynamic performance tables.
○ Dictionary Objects
Objects in 'SYS', 'SYSTEM' and all non-user defined schemas.

TIP The Objects step will be skipped when Database, Fixed Objects or Dictionary Objects is selected.

Gathering Optimizer Statistics Manually

You may need to gather statistics manually at certain times, such as when the contents of a table have changed so much between automatic gathering jobs that the statistics no longer represent the table accurately. This is common for large tables that experience more than a 10 percent change in size in a 24-hour period.

Best practice tip: Collect statistics often enough that the table never changes by more than about 10 percent between collection periods. This may require manual statistics collection or additional maintenance windows.

Statistics can be manually collected by using either Enterprise Manager or the DBMS_STATS package. System statistics can be gathered only by using the DBMS_STATS package.

The Gather Optimizer Statistics menu selection starts a wizard that allows you to select the scope, objects, options, and schedule for a job that will gather optimizer statistics. The wizard submits a DBMS_STATS.GATHER_*_STATS job at the scope you specify: table, schema, or database. In this wizard, you set the preferences for the default values used by the DBMS_STATS package and you schedule this job to run at a time that you determine.

Gathering statistics manually is not recommended because the statistics are gathered more efficiently and with less impact on users during the maintenance windows.

A manual job can also be submitted if the automatic job has failed or been disabled.

Gathering Optimizer Statistics Manually (continued)

You can also gather optimizer statistics with the DBMS_STATS package directly:

```
SQL> EXEC dbms_stats.gather_table_stats('HR','EMPLOYEES');
SQL> SELECT num_rows FROM dba_tables
  2  WHERE owner='HR' AND table_name = 'EMPLOYEES';
  NUM_ROWS
----------
       214
```

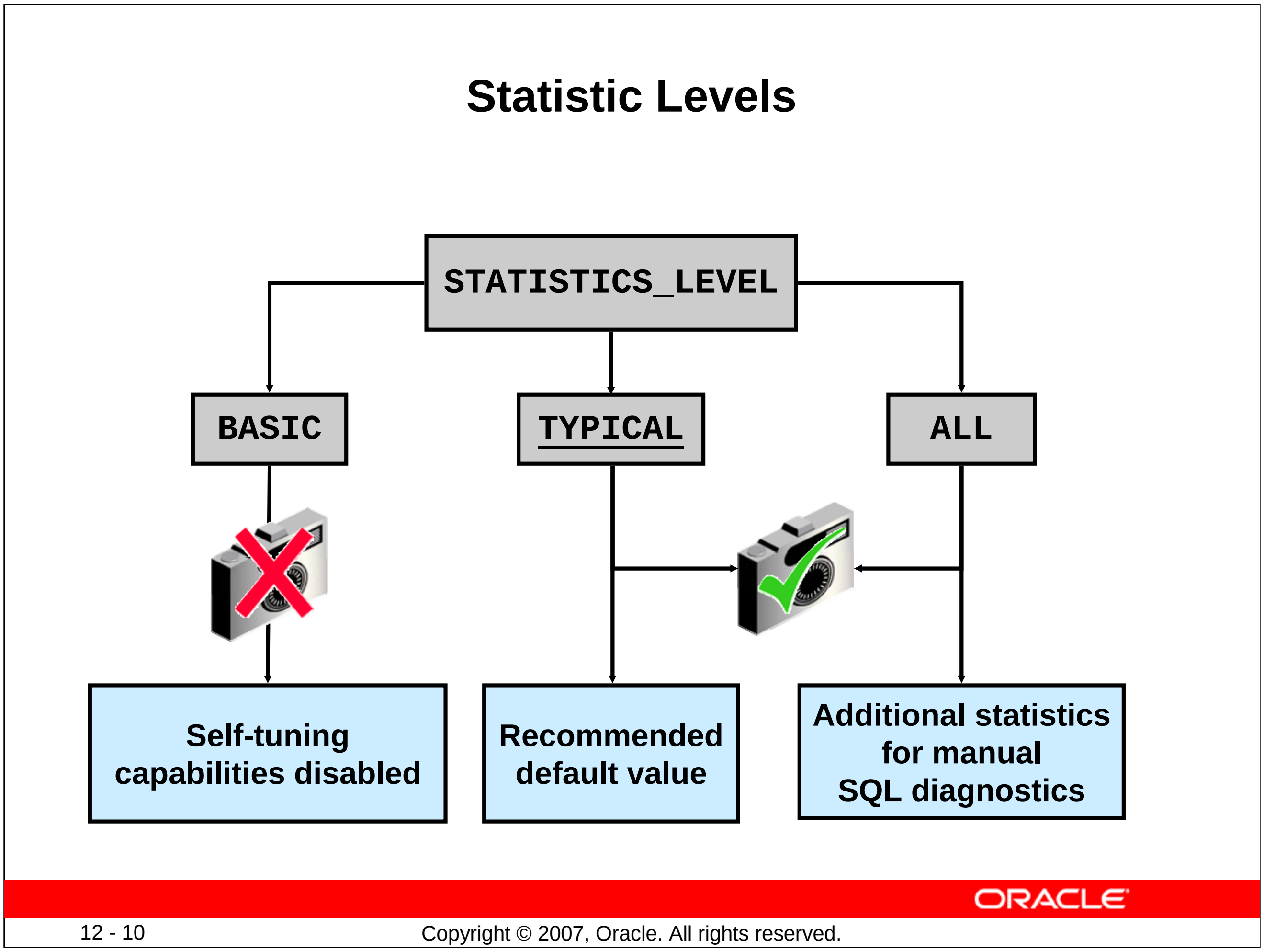
Notice that the number of rows now correctly reflects what was in the table at the time that the statistics were gathered. DBMS_STATS also enables manual collection of statistics for an entire schema or even for the whole database.

System statistics do not change unless the workload significantly changes. As a result, system statistics do not need frequent adjustment. The DBMS_STATS.GATHER_SYSTEM_STATS procedure will collect system statistics over a specified period, or you can start the gathering of system statistics and make another call to stop gathering.

Best practice tip: Use the following command when you create a database:

```
SQL> EXEC dbms_stats.gather_system_stats('NOWORKLOAD');
```

The NOWORKLOAD option takes a few minutes (depending on the size of the database) and captures estimates of I/O characteristics such as average read seek time and I/O transfer rate.



Statistic Levels

STATISTICS_LEVEL

BASIC

TYPICAL

ALL

Self-tuning capabilities disabled

Recommended default value

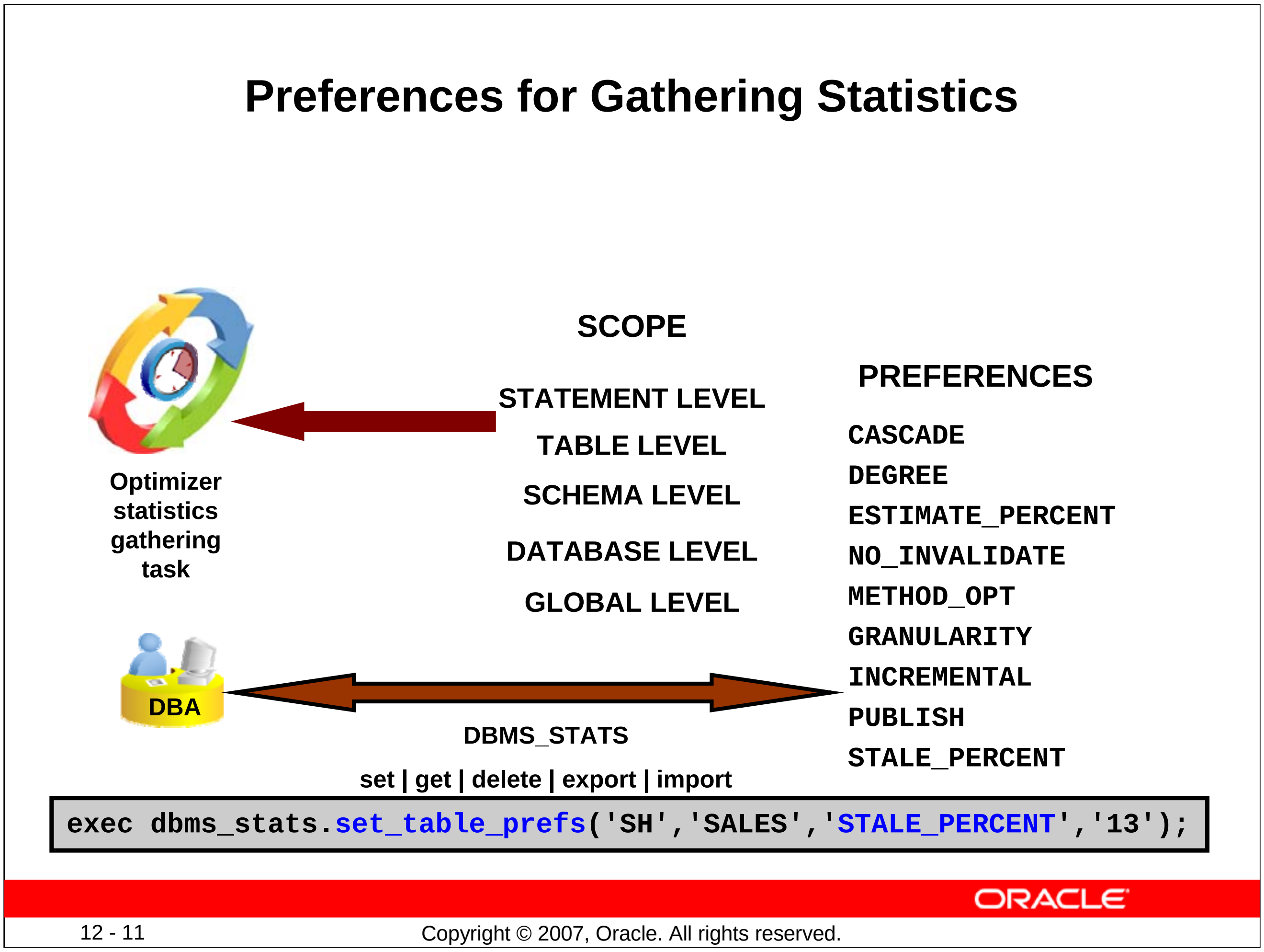
Additional statistics for manual SQL diagnostics

Statistic Levels

The `STATISTICS_LEVEL` initialization parameter controls the capture of a variety of statistics and various advisors, including the automatic maintenance tasks. The automatic maintenance tasks include gathering optimizer statistics. The `STATISTICS_LEVEL` parameter can be set to the following levels:

- **`BASIC`:** The computation of AWR statistics and metrics is turned off. The automatic optimizer statistics task is disabled, as are all advisors and server-generated alerts.
- **`TYPICAL`:** Major statistics are collected that are required for database self-management. They represent what is typically needed to monitor Oracle database behavior. This includes automatic gathering of statistics to reduce the likelihood of poorly performing SQL statements due to stale or invalid statistics.
- **`ALL`:** All possible statistics are captured. This level of capture adds timed OS statistics and plan execution statistics. These statistics are not needed in most cases and should not be enabled for best performance; they are sometimes needed for specific diagnostics tests.

Oracle recommends that the default value of `TYPICAL` be set for the `STATISTICS_LEVEL` initialization parameter. Setting the value to `BASIC` disables the automatic gathering of optimizer statistics.



Preferences for Gathering Statistics

Optimizer statistics gathering task

SCOPE

STATEMENT LEVEL

TABLE LEVEL

SCHEMA LEVEL

DATABASE LEVEL

GLOBAL LEVEL

PREFERENCES

CASCADE
DEGREE
ESTIMATE_PERCENT
NO_INVALIDATE
METHOD_OPT
GRANULARITY
INCREMENTAL
PUBLISH
STALE_PERCENT

DBA

DBMS_STATS

set | get | delete | export | import

```
exec dbms_stats.set_table_prefs('SH','SALES','STALE_PERCENT','13');
```

Preferences for Gathering Statistics

The DBMS_STATS.GATHER_*_STATS procedures can be called at various levels to gather statistics for an entire database or for individual objects such as tables. When the GATHER_*_STATS procedures are called, several of the parameters are often allowed to default. The supplied defaults work well for most of the objects in the database, but for some objects or schemas the defaults need to be changed. Instead of running manual jobs for each of these objects, Oracle Database 11*g* allows you to set values (called *preferences*) for individual objects, schemas, or databases, or to change the default values with the global-level command.

The preferences specify the parameters that are given to the gather procedures. The SET_*_PREFS procedures create preference values for any object that is not owned by SYS or SYSTEM. The expected use is that the DBA will set the global preferences for any parameters that should be database-wide. These will be applied for any parameter that is allowed to default.

The SET_DATATBASE_PREFS procedure iterates over all the tables and schemas in the database setting the specified preference. SET_SCHEMA_PREFS iterates over the tables in the specified schema. SET_TABLE_PREFS sets the preference value for a single table.

All object preferences—whether set at the database, schema, or table level—are held in a single table. Changing the preferences at the schema level overwrites the preferences that were previously set at the table level.

Preferences for Gathering Statistics (continued)

When the various gather procedures execute, they retrieve the object-level preferences that were set for each object. You can view the object-level preferences in the DBA_TAB_STAT_PREFS view. Any preferences that are not set at the object level will be set to the global-level preferences. You can see the global preferences by calling the DBMS_STATS.GET_PREFS procedure for each preference.

You can set, get, delete, export, and import those preferences at the table, schema, database, and global levels. The preference values are expected to be set from global to table levels, applying the preferences to the smallest group last.

Preferences in Oracle Database 11*g*:

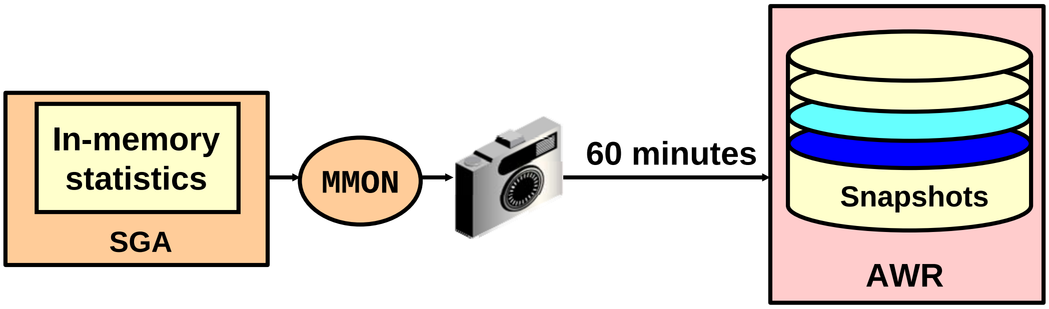
- CASCADE determines whether index statistics are collected as part of gathering table statistics.
- DEGREE sets the degree of parallelism that is used for gathering statistics.
- PUBLISH is used to decide whether to publish the statistics to the dictionary or store them in a private area. This enables the DBA to validate the statistics before publishing them to the data dictionary with the PUBLISH_PENDING_STATS procedure.
- STALE_PERCENT is used to determine the threshold level at which an object is considered to have stale statistics. The value is a percentage of the rows modified since the last statistics gathering. The example changes the 10 percent default to 13 percent for SH.SALES only.
- INCREMENTAL is used to gather global statistics on partitioned tables in an incremental way.
- METHOD_OPT determines the columns and histogram parameters that are used to gather column statistics.
- GRANULARITY determines the granularity of statistics to collect (which is pertinent only if the table is partitioned).
- NO_INVALIDATE is used to determine whether to invalidate cursors.
- ESTIMATE_PERCENT is used to determine the number of rows to sample to obtain good statistics. It is a percentage of the number of rows in the table.

Note: For details about these preferences, see the DBMS_STATS documentation in the *Oracle Database PL/SQL Packages and Types Reference*.

Preferences may be deleted with the DBMS_STATS.DELETE_*_PREFS procedures at the table, schema, and database levels. You can reset the global preferences to the recommended values with the DBMS_STATS.RESET_PARAM_DEFAULTS procedure.

Automatic Workload Repository (AWR)

- Built-in repository of performance information
- Snapshots of database metrics taken every 60 minutes and retained for eight days
- Foundation for all self-management functions



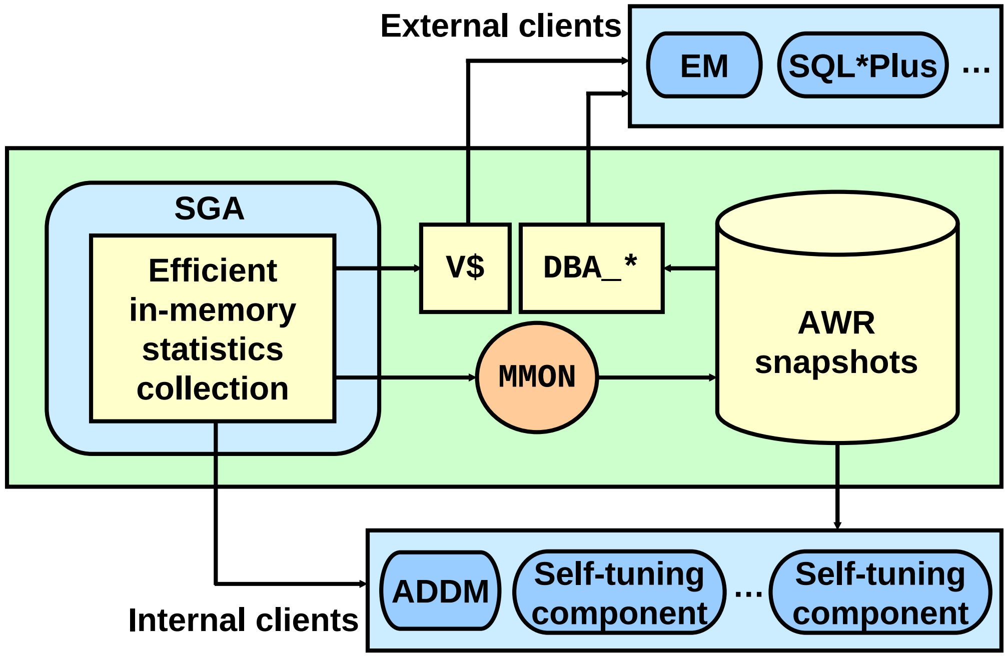
Automatic Workload Repository (AWR)

The AWR is the infrastructure that provides services to Oracle Database 11*g* components to collect, maintain, and utilize statistics for problem detection and self-tuning purposes. You can view it as a data warehouse for database statistics, metrics, and so on.

Every 60 minutes (by default) the database automatically captures statistical information from the SGA and stores it in the AWR in the form of snapshots. These snapshots are stored on the disk by a background process called Manageability Monitor (`MMON`). By default, snapshots are retained for eight days. You can modify both the snapshot interval and the retention intervals.

The AWR contains hundreds of tables, all belonging to the `SYSMAN` schema and stored in the `SYSAUX` tablespace. The Oracle database does not support direct SQL access to the repository. Instead, use Enterprise Manager or the `DBMS_WORKLOAD_REPOSITORY` package to work with the AWR.

AWR Infrastructure



AWR Infrastructure

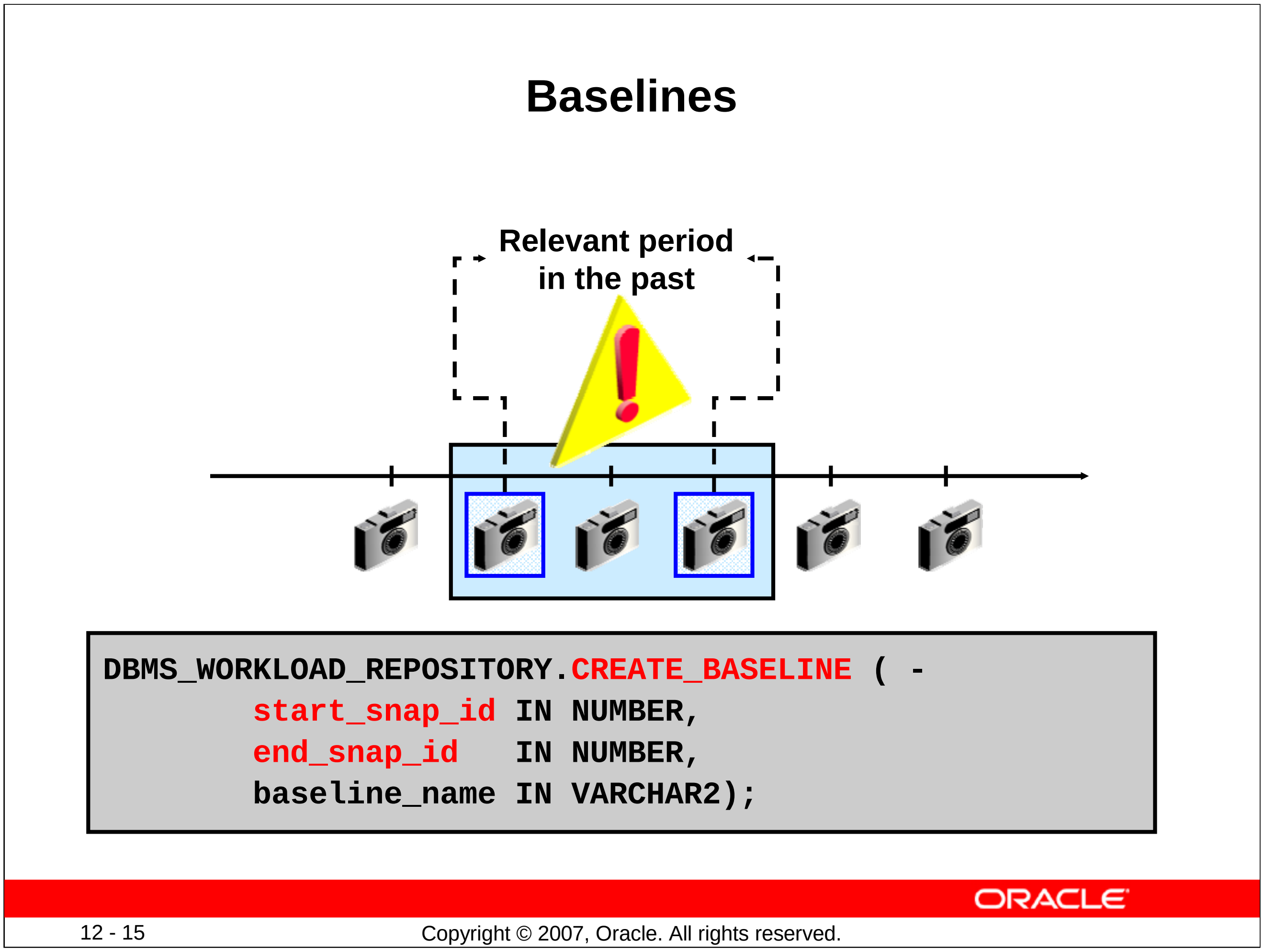
The AWR infrastructure has two major parts:

- An in-memory statistics collection facility that is used by Oracle Database 11*g* components to collect statistics. These statistics are stored in memory for performance reasons. Statistics stored in memory are accessible through dynamic performance (V$) views.
- The AWR snapshots that represent the persistent portion of the facility. AWR snapshots are accessible through data dictionary views and Enterprise Manager Database Control.

Statistics are stored in persistent storage for several reasons:

- The statistics need to survive instance crashes.
- Some analyses need historical data for baseline comparisons.
- A memory overflow can occur. When old statistics are replaced by new ones because of memory shortage, the replaced data can be stored for later use.

The memory version of the statistics is transferred to disk on a regular basis by the `MMON` background process. With the AWR, the Oracle database provides a way to capture historical statistics data automatically without DBA intervention.



Baselines

```
DBMS_WORKLOAD_REPOSITORY.CREATE_BASELINE ( -
        start_snap_id IN NUMBER,
        end_snap_id   IN NUMBER,
        baseline_name IN VARCHAR2);
```

Baselines

A baseline is an AWR snapshot set that you have tagged for important periods. A snapshot set is defined on a pair of snapshots; the snapshots are identified by their snapshot sequence numbers (`snap_id`). Each snapshot set corresponds to one and only one pair of snapshots.

A snapshot set can be identified by either a user-supplied name or a system-generated identifier. You create a snapshot set by executing the `DBMS_WORKLOAD_REPOSITORY.CREATE_BASELINE` procedure and specifying a name and a pair of snapshot identifiers. A snapshot set identifier is assigned to the newly created snapshot set. Snapshot set identifiers are unique for the life of a database.

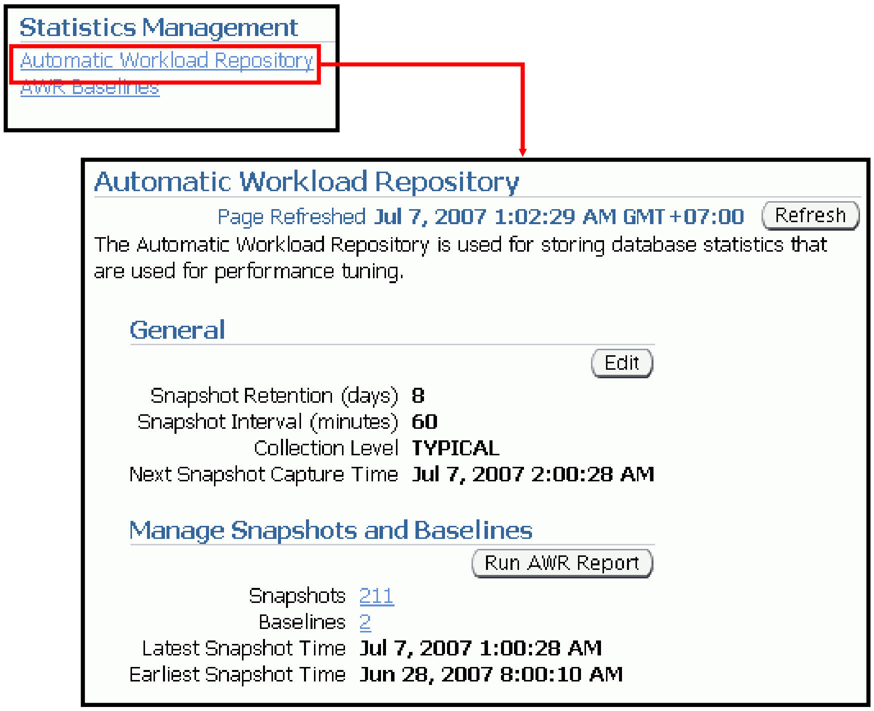
Snapshot sets are used to retain snapshot data. Snapshots belonging to snapshot sets are retained until the snapshot sets are dropped.

You set up snapshot sets, usually from some representative periods in the past, to be used for comparisons with current system behavior. You can also set up threshold-based alerts by using snapshot sets from Enterprise Manager Database Control.

You can get `snap_ids` directly from `DBA_HIST_SNAPSHOT` or Database Control.

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

Enterprise Manager and the AWR



Automatic Workload Repository

Page Refreshed **Jul 7, 2007 1:02:29 AM GMT+07:00** Refresh

The Automatic Workload Repository is used for storing database statistics that are used for performance tuning.

General

Edit

Snapshot Retention (days) **8**
Snapshot Interval (minutes) **60**
Collection Level **TYPICAL**
Next Snapshot Capture Time **Jul 7, 2007 2:00:28 AM**

Manage Snapshots and Baselines

Run AWR Report

Snapshots 211
Baselines 2
Latest Snapshot Time **Jul 7, 2007 1:00:28 AM**
Earliest Snapshot Time **Jun 28, 2007 8:00:10 AM**

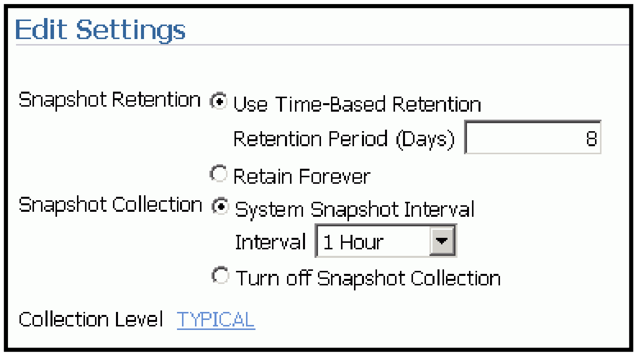
Enterprise Manager and the AWR

Click the Server tab, and then click Automatic Workload Repository in the Statistics Management section. On the Automatic Workload Repository page, click Edit to change the settings.

From the Automatic Workload Repository page, you can:

- Edit the workload repository settings
- Look at the detailed information about created snapshots and manually create new ones
- Create AWR baselines.
- Generate an AWR report

Managing the AWR



Edit Settings

Snapshot Retention ● Use Time-Based Retention

Retention Period (Days) 8

○ Retain Forever

Snapshot Collection ● System Snapshot Interval

Interval 1 Hour

○ Turn off Snapshot Collection

Collection Level TYPICAL

- **Retention period**
  - **Default: Eight days**
  - **Consider storage needs**
- **Collection interval**
  - **Default: 60 minutes**
  - **Consider storage needs and performance impact**
- **Collection level**
  - **Basic (disables most ADDM functionality)**
  - **Typical (recommended)**
  - **All (adds additional SQL tuning information to snapshots)**

Managing the AWR

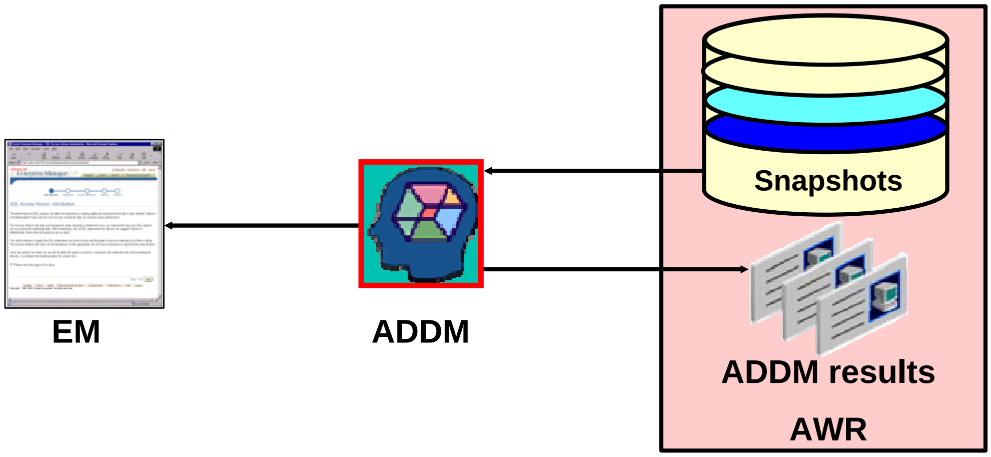
AWR settings include retention period, collection interval, and collection level. Remember that decreasing any of these settings affects the functionality of components that depend on the AWR, including the advisors.

Increasing the settings can provide improved advisor recommendations—but at the cost of the space that is required to store the snapshots and the performance expended in collecting the snapshot information.

Consider setting collection level to `ALL` when tuning a new application. The `ALL` setting collects SQL execution plans and timing statistics that enhance the recommendations of the SQL advisors. When tuning is complete, this setting should be returned to the `TYPICAL` setting.

Automatic Database Diagnostic Monitor (ADDM)

- Runs after each AWR snapshot
- Monitors the instance; detects bottlenecks
- Stores results in the AWR



Automatic Database Diagnostic Monitor (ADDM)

Unlike the other advisors, the ADDM runs automatically after each AWR snapshot. Each time a snapshot is taken, the ADDM performs an analysis of the period corresponding to the last two snapshots. The ADDM proactively monitors the instance and detects most bottlenecks before they become a significant problem.

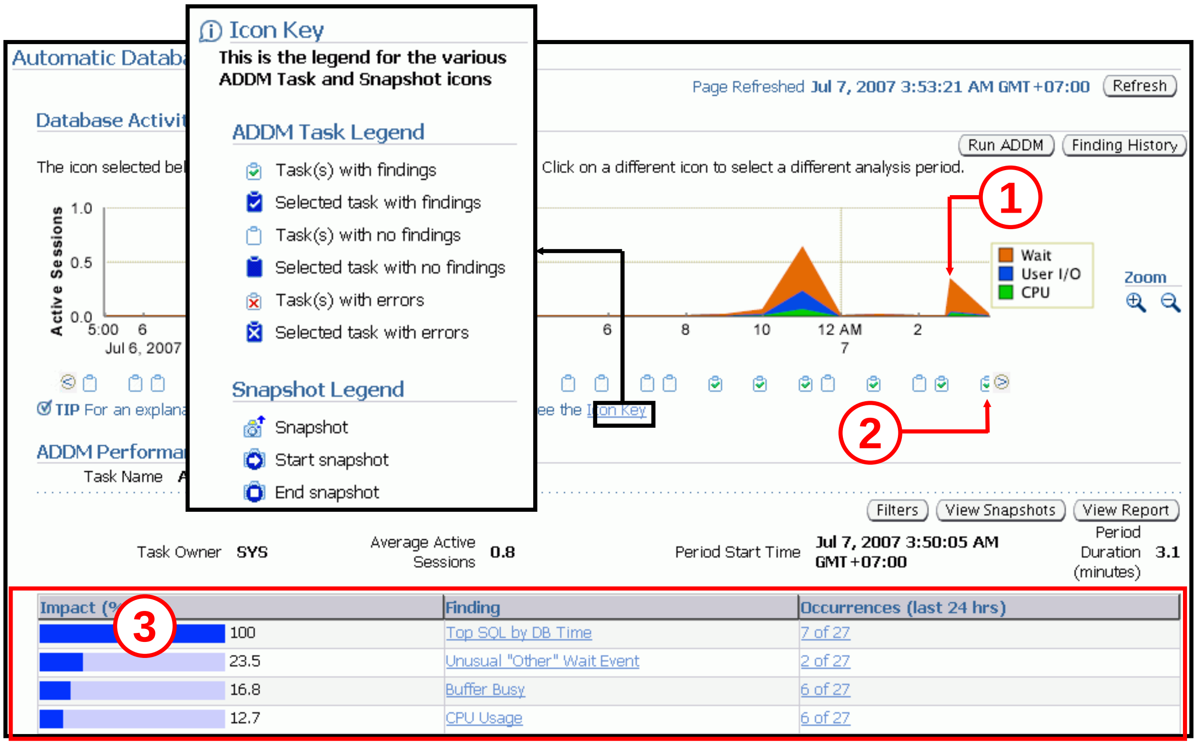
In many cases, the ADDM recommends solutions for detected problems and even quantifies the benefits for the recommendations.

Some common problems that are detected by the ADDM:

- CPU bottlenecks
- Poor Oracle Net connection management
- Lock contention
- Input/output (I/O) capacity
- Undersizing of database instance memory structures
- High-load SQL statements
- High PL/SQL and Java time
- High checkpoint load and cause (for example, small log files)

The results of each ADDM analysis are stored in the AWR and are also accessible through Enterprise Manager.

ADDM Findings



This is the legend for the various ADDM Task and Snapshot icons

Page Refreshed Jul 7, 2007 3:53:21 AM GMT+07:00

Click on a different icon to select a different analysis period.

Task Owner	SYS	Average Active Sessions	0.8	Period Start Time	Jul 7, 2007 3:50:05 AM GMT+07:00
Period Duration (minutes)	3.1				

Impact (%		Finding	Occurrences (last 24 hrs)
	100	Top SQL by DB Time	7 of 27
	23.5	Unusual "Other" Wait Event	2 of 27
	16.8	Buffer Busy	6 of 27
	12.7	CPU Usage	6 of 27

ADDM Findings

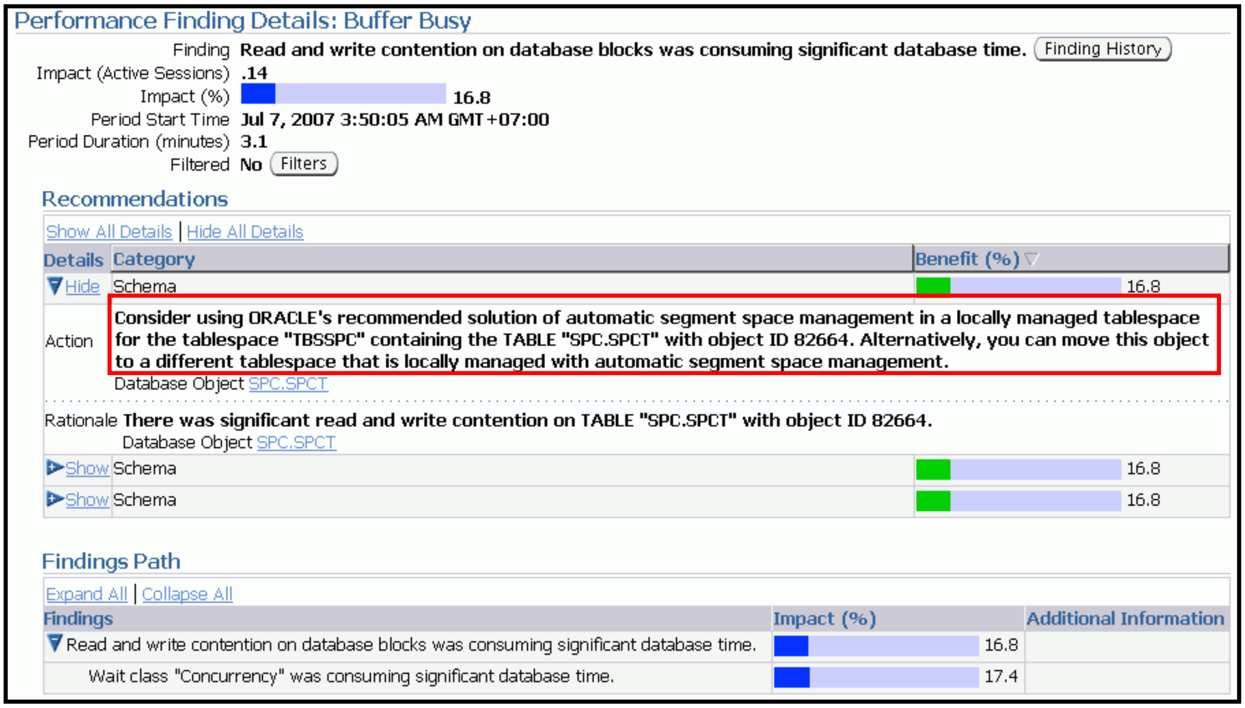
On the Automatic Database Diagnostic Monitor (ADDM) page, you see the detailed findings for the latest ADDM run. Database Time represents the sum of the nonidle time spent by sessions in the database for the analysis period. A specific impact percentage is given for each finding. The impact represents the time consumed by the corresponding issue compared with the database time for the analysis period.

In the slide, note the following:

1. The graphic shows that the number of average active users increased dramatically at this point. In addition, the major problem was a `Wait` problem.
2. The icon shows that the ADDM output displayed at the bottom of the page corresponds to this point in time. You can go into the past (to view previous analyses) by clicking the other icons.
3. The findings give you a short summary of what the ADDM found as tunable areas. By clicking a particular issue, you are directed to the Performance Finding Details page.

Click the View Report button to get details about the performance analysis in the form of a text report.

ADDM Recommendations



Performance Finding Details: Buffer Busy

Finding	Read and write contention on database blocks was consuming significant database time. Finding History
Impact (Active Sessions)	.14
Impact (%)	16.8
Period Start Time	Jul 7, 2007 3:50:05 AM GMT+07:00
Period Duration (minutes)	3.1
Filtered	No Filters

Recommendations

Show All Details | Hide All Details

Details	Category	Benefit (%) ▽
▼Hide	Schema	16.8
Action	Consider using ORACLE's recommended solution of automatic segment space management in a locally managed tablespace for the tablespace "TBSSPC" containing the TABLE "SPC.SPCT" with object ID 82664. Alternatively, you can move this object to a different tablespace that is locally managed with automatic segment space management. Database Object SPC.SPCT	
Rationale	There was significant read and write contention on TABLE "SPC.SPCT" with object ID 82664. Database Object SPC.SPCT	
▶Show	Schema	16.8
▶Show	Schema	16.8

Findings Path

Expand All | Collapse All

Findings	Impact (%)	Additional Information
▼Read and write contention on database blocks was consuming significant database time.	16.8	
Wait class "Concurrency" was consuming significant database time.	17.4	

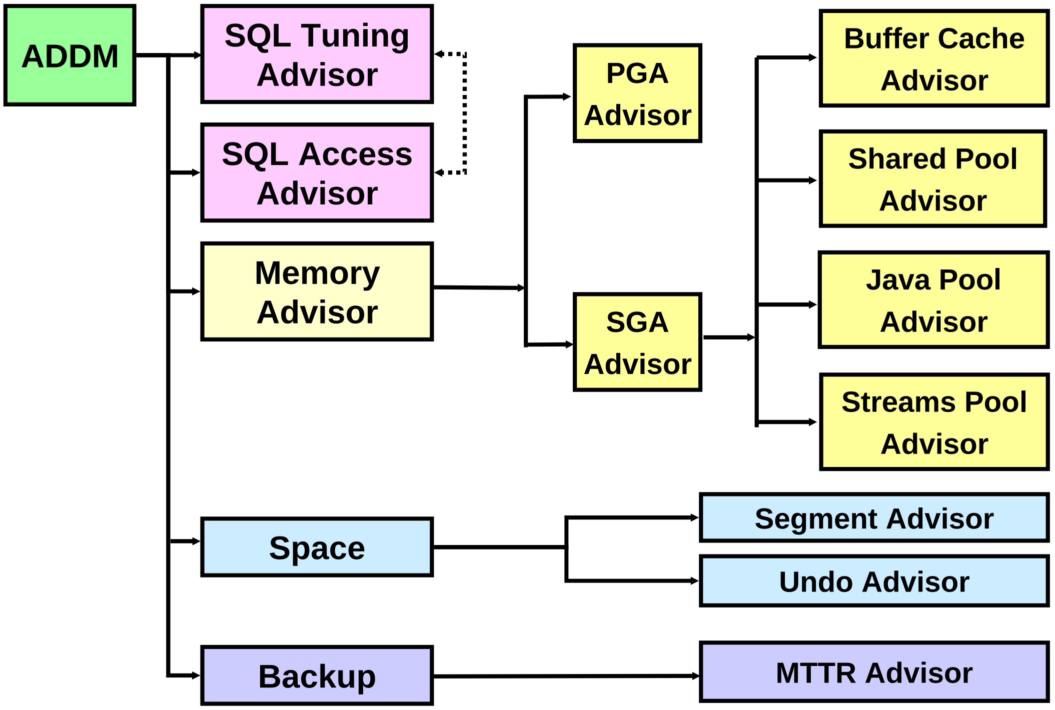
ADDM Recommendations

On the Performance Finding Details page, you are given recommendations for solving the corresponding issue. Recommendations are grouped into Schema, SQL Tuning, Database Configuration, and many other categories. The `Benefit(%)`column gives you the maximum reduction in database elapsed time if the recommendation is implemented.

The ADDM considers a variety of changes to a system. Its recommendations can include:

- **Hardware changes:** Adding CPUs or changing the I/O subsystem configuration
- **Database configuration:** Changing initialization parameter settings
- **Schema changes:** Hash-partitioning a table or index, or using Automatic Segment Space Management (ASSM)
- **Application changes:** Using the cache option for sequences, or using bind variables
- **Using other advisors:** Running the SQL Tuning Advisor on high-load SQL, or running the Segment Advisor on hot objects

Advisory Framework



ADDM

SQL Tuning Advisor

SQL Access Advisor

Memory Advisor

PGA Advisor

SGA Advisor

Buffer Cache Advisor

Shared Pool Advisor

Java Pool Advisor

Streams Pool Advisor

Space

Segment Advisor

Undo Advisor

Backup

MTTR Advisor

Advisory Framework

Advisors provide you with useful feedback about resource utilization and performance for their respective server components. For example, the Memory Advisor provides a recommended value for the `MEMORY_TARGET` initialization parameter, which controls the total amount of memory used by the Oracle database instance.

By building on the data captured in the AWR, the ADDM enables the Oracle database to diagnose its own performance and determine how identified problems can be resolved. ADDM runs automatically after each AWR statistics capture. It can potentially call other advisors.

Here are the major benefits that are provided by the advisor infrastructure:

- All advisors use a uniform interface.
- All advisors have a common data source and results storage by using the workload repository.

Not all advisors are shown in the slide (for example, the Data Recovery Advisor and the SQL Repair Advisor are not listed).

Advisory Framework (continued)

Automatic Database Diagnostic Monitor (ADDM)

The ADDM is a server-based expert that reviews database performance every 60 minutes. Its goal is to detect possible system bottlenecks early and recommend fixes before system performance degrades noticeably.

Memory Advisors

The Memory Advisor is actually a collection of several advisory functions that help determine the best settings for the total memory used by the database instance. The System Global Area (SGA) has a set of advisors for the shared pool, database buffer cache, Java pool, and streams pool. The Java pool and streams pool advisors are not exposed on the EM Memory Advisor page. There is an advisor for the Program Global Area (PGA). In addition to the advisory functions, this page provides a central point of control for the large pool and the Java pool.

Mean-Time-To-Recover (MTTR) Advisor

Using the MTTR Advisor, you set the length of time required for the database to recover after an instance crash.

Segment Advisor

This advisor looks for tables and indexes that consume more space than they require. The advisor checks for inefficient space consumption at the tablespace or schema level and produces scripts to reduce space consumption where possible.

SQL Access Advisor

This advisor analyzes all SQL statements that are issued in a given period and suggests the creation of additional indexes or materialized views that will improve performance.

SQL Tuning Advisor

This advisor analyzes an individual SQL statement and makes recommendations for improving its performance. Recommendations may include actions such as rewriting the statement, changing the instance configuration, or adding indexes. The SQL Tuning Advisor is not invoked directly. Instead, it is called from within other tools (such as Top SQL or Top Sessions) to help optimize high-impact SQL statements.

Undo Management Advisor

With the Undo Management Advisor, you can determine the undo tablespace size that is required to support a given retention period. Undo management and the use of the advisor are covered in the lesson titled “Managing Undo Data.”

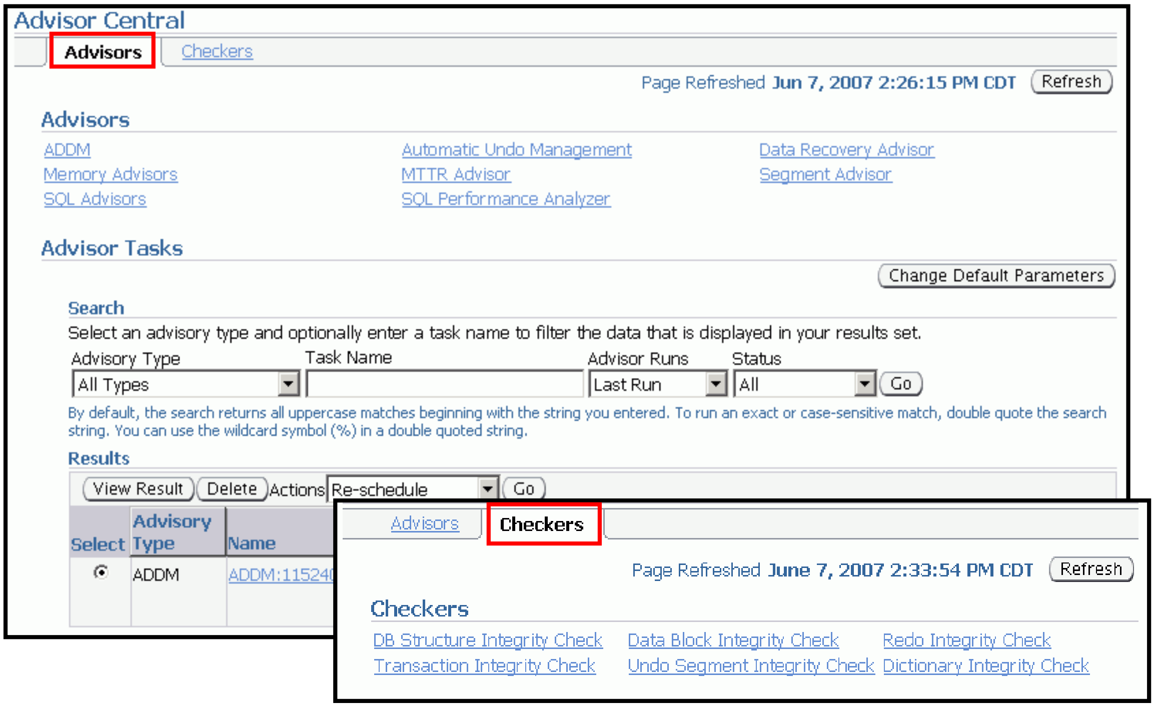
Data Recovery Advisor

This advisor automatically diagnoses persistent data failures, presents repair options to the user, and executes repairs at the user’s request. The purpose of the Data Recovery Advisor is to reduce the mean time to recover (MTTR) and provide a centralized tool for automated data repair.

SQL Repair Advisor

You run the SQL Repair Advisor after a SQL statement fails with a critical error that generates a problem in the Automatic Diagnostic Repository. The advisor analyzes the statement and, in many cases, recommends a patch to repair the statement. If you implement the recommendation, the applied SQL patch circumvents the failure by causing the query optimizer to choose an alternative execution plan for future executions. This is done without changing the SQL statement itself.

Enterprise Manager and Advisors



Page Refreshed Jun 7, 2007 2:26:15 PM CDT

Advisors

ADDM, Memory Advisors, SQL Advisors, Automatic Undo Management, MTTR Advisor, SQL Performance Analyzer, Data Recovery Advisor, Segment Advisor

Advisor Tasks

Search

Select an advisory type and optionally enter a task name to filter the data that is displayed in your results set.

Advisory Type: All Types; Task Name: ; Advisor Runs: Last Run; Status: All

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Results

Actions: Re-schedule

Select	Advisory Type	Name
●	ADDM	ADDM:11524

Page Refreshed June 7, 2007 2:33:54 PM CDT

Checkers

DB Structure Integrity Check, Transaction Integrity Check, Data Block Integrity Check, Undo Segment Integrity Check, Redo Integrity Check, Dictionary Integrity Check

Enterprise Manager and Advisors

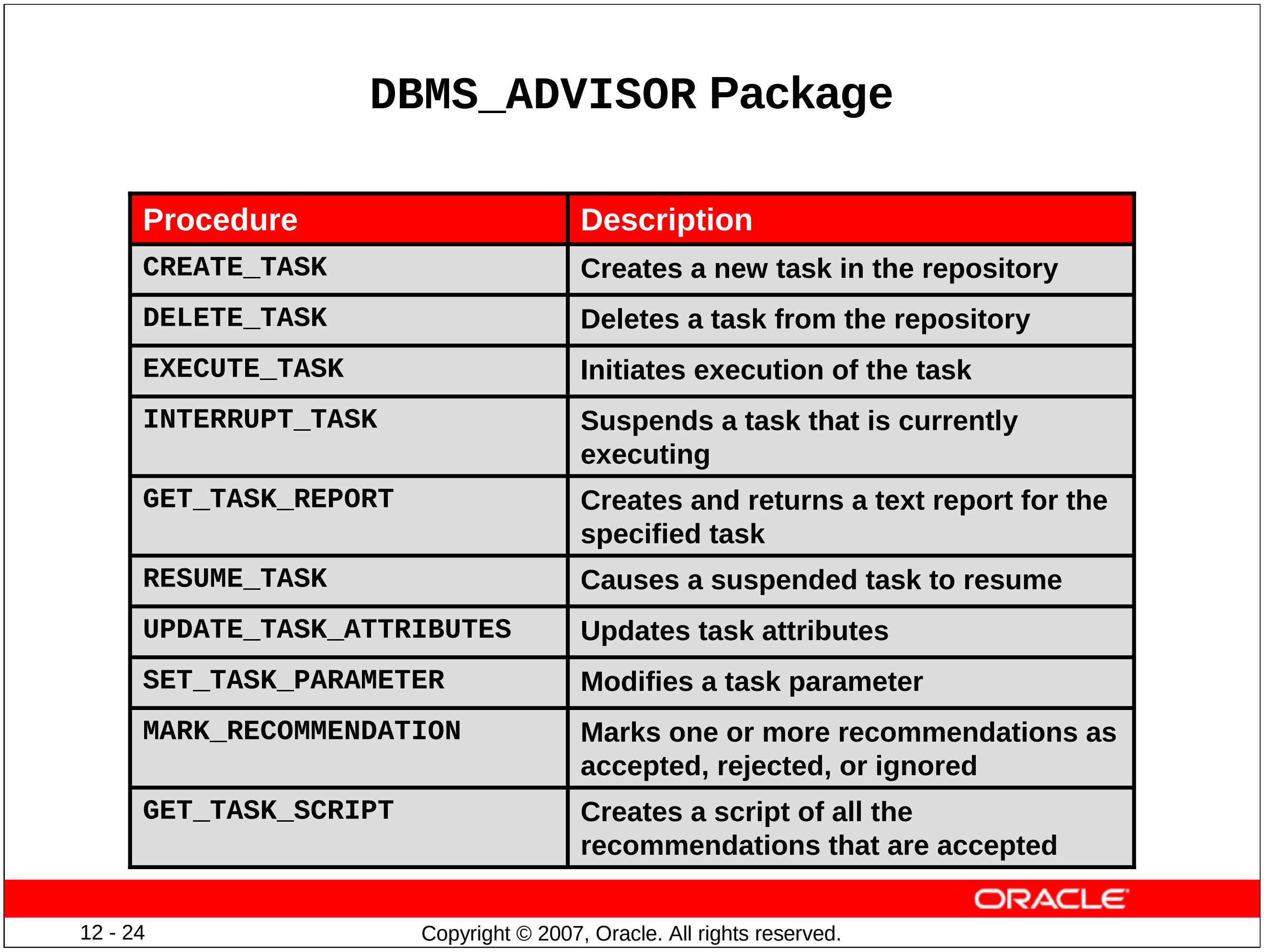
The Advisor Central page is the main page of all advisors. You can reach this page by clicking the Advisor Central link in the list of Related Links on the Database Control Home page. However, this is not the only place in Database Control where advisors can be invoked. It is also possible to have access to advisors in certain contexts.

On the Advisors tab of the Advisor Central page, you can list all the advisor tasks that are registered in the workload repository. You can also filter this list by advisor type and for predefined time periods.

The Checkers tab of the Advisor Central page enables you to schedule various database integrity checkers. You can list all the checker runs by name, type, or time period.

Some advisors are described in greater detail in the lessons titled “Managing Undo Data,” “Performance Management,” and “Backup and Recovery Concepts.”

Note: Use the Change Default Parameters page to change the default expiration (in days) for all future tasks. You can also use this page to change the parameters of some important advisors.



DBMS_ADVISOR Package

Procedure	Description
CREATE_TASK	Creates a new task in the repository
DELETE_TASK	Deletes a task from the repository
EXECUTE_TASK	Initiates execution of the task
INTERRUPT_TASK	Suspends a task that is currently executing
GET_TASK_REPORT	Creates and returns a text report for the specified task
RESUME_TASK	Causes a suspended task to resume
UPDATE_TASK_ATTRIBUTES	Updates task attributes
SET_TASK_PARAMETER	Modifies a task parameter
MARK_RECOMMENDATION	Marks one or more recommendations as accepted, rejected, or ignored
GET_TASK_SCRIPT	Creates a script of all the recommendations that are accepted

DBMS_ADVISOR Package

The DBMS_ADVISOR package contains all constants and procedure declarations for all advisor modules. You can use this package to execute tasks from the command line.

To execute advisor procedures, you must be granted the ADVISOR privilege. The ADVISOR privilege permits full access to the advisor procedures and views.

Note: For more information about all the procedures found in the DBMS_ADVISOR package, see the *Oracle Database PL/SQL Packages and Types Reference*.

Automated Maintenance Tasks

Autotask maintenance process:

1. Maintenance Window opens.
2. Autotask background process schedules jobs.
3. Scheduler initiates jobs.
4. Resource Manager limits impact of Autotask jobs.

Default Autotask maintenance jobs:

- Gathering optimizer statistics
- Automatic Segment Advisor
- Automatic SQL Advisor

Automated Maintenance Tasks

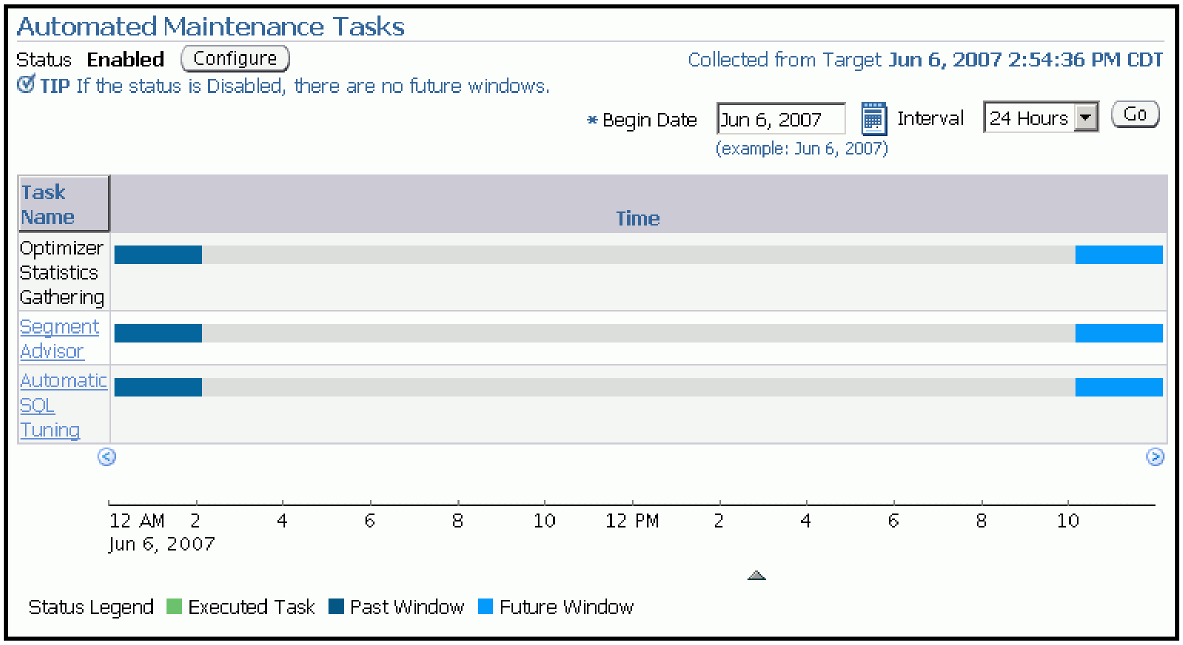
By analyzing the information stored in the AWR, the database can identify the need to perform routine maintenance tasks, such as optimizer statistics refresh. The automated maintenance tasks infrastructure enables the Oracle database to automatically perform such operations. It uses the Scheduler to run such tasks in predefined maintenance windows.

By default, the weekday maintenance windows start at 10:00 PM and lasts 4 hours. On Saturday and Sunday, the maintenance window starts at 6:00 AM and lasts for 20 hours. All attributes of the maintenance windows are customizable, including the start and end time, frequency, days of the week, and so on. In addition, the impact of automated maintenance tasks on normal database operations can be limited by associating a Database Resource Manager resource plan to the maintenance window.

Examples of maintenance:

- Optimizer statistics are automatically refreshed by using the automatic maintenance task infrastructure.
- The Automatic Segment Advisor has default jobs, which run in the maintenance window.
- When creating a database with the DBCA, you can initiate regular database backups.

Automated Maintenance Tasks



Automated Maintenance Tasks

Status **Enabled** Configure

Collected from Target **Jun 6, 2007 2:54:36 PM CDT**

✓ **TIP** If the status is Disabled, there are no future windows.

* Begin Date Jun 6, 2007 Interval 24 Hours Go
(example: Jun 6, 2007)

Task Name	Time
Optimizer Statistics Gathering	
Segment Advisor	
Automatic SQL Tuning	

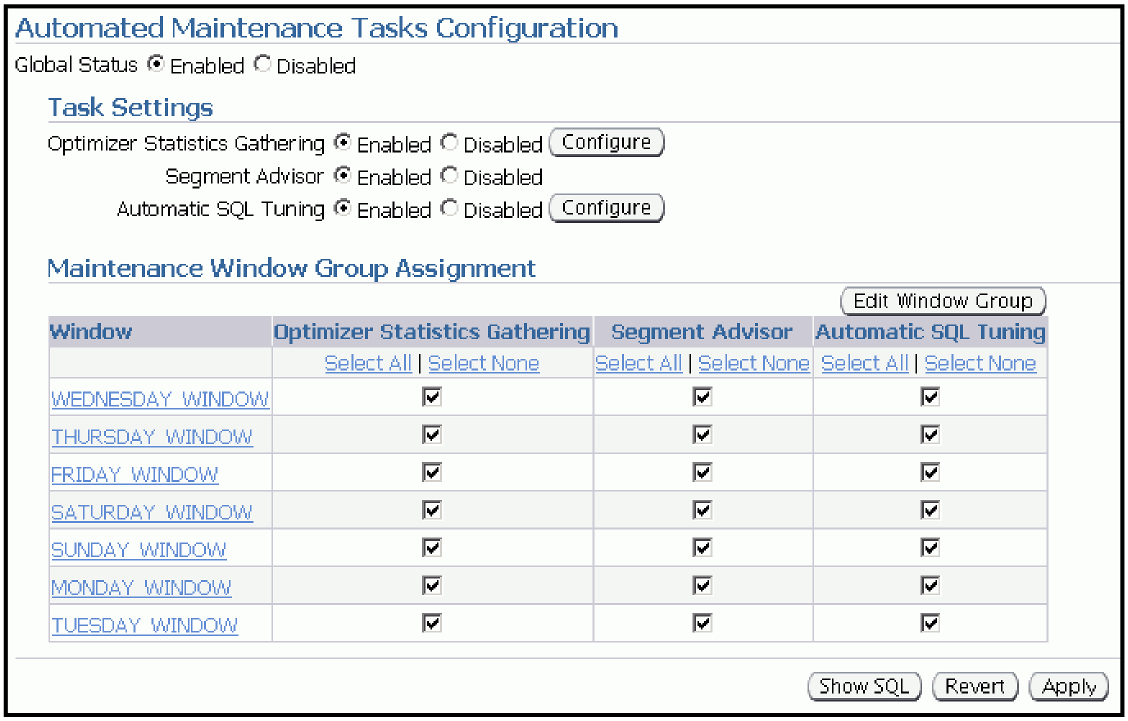
Status Legend 🟩 Executed Task ■ Past Window ■ Future Window

Automated Maintenance Tasks (continued)

Click Automated Maintenance Tasks under the Scheduler heading on the Server page to access the Automated Maintenance Task page, where you can view the automated maintenance task schedule and recent history. From here you can drill down to details on some tasks. Click Configure to go to the Automated Maintenance Tasks Configuration page. A task executes in a window. The graph shows the last window in which a task was executed and the next window in which the task is scheduled to be executed.

Note: The default windows for tasks are shown in the example. When the maintenance window closes, the Scheduler terminates the optimizer statistics gathering job by default. The remaining objects are then processed in the next maintenance window.

Automated Maintenance Tasks Configuration



Automated Maintenance Tasks Configuration

Global Status ● Enabled ○ Disabled

Task Settings

Optimizer Statistics Gathering ● Enabled ○ Disabled

Segment Advisor ● Enabled ○ Disabled

Automatic SQL Tuning ● Enabled ○ Disabled

Maintenance Window Group Assignment

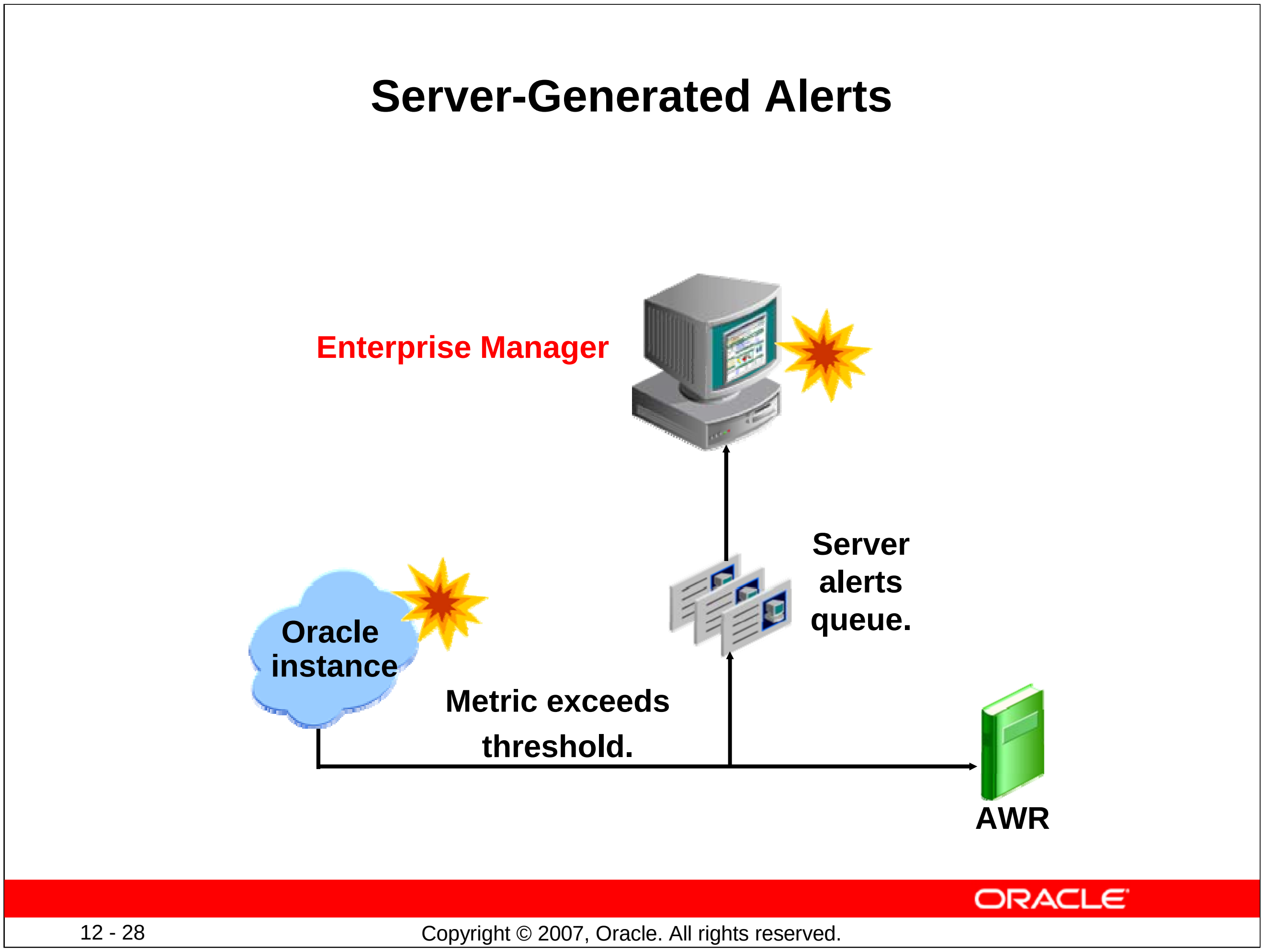
Window	Optimizer Statistics Gathering	Segment Advisor	Automatic SQL Tuning
	Select All | Select None	Select All | Select None	Select All | Select None
WEDNESDAY_WINDOW	☑	☑	☑
THURSDAY_WINDOW	☑	☑	☑
FRIDAY_WINDOW	☑	☑	☑
SATURDAY_WINDOW	☑	☑	☑
SUNDAY_WINDOW	☑	☑	☑
MONDAY_WINDOW	☑	☑	☑
TUESDAY_WINDOW	☑	☑	☑

Automated Maintenance Tasks Configuration

On the Automated Maintenance Tasks Configuration page, you can enable and disable automatic maintenance tasks—all at once, by individual tasks, or by particular windows. You can also configure the settings that are used for optimizer statistics gathering and the job control parameters for the automatic SQL Tuning Advisor.

Select the window name to view or edit the window schedule.

Click Edit Window Group to add and remove windows in the window group.



Server-Generated Alerts

12 - 28

Server-Generated Alerts

Alerts are notifications of when a database is in an undesirable state and needs your attention. By default, the Oracle database provides alerts via Enterprise Manager Database Control. Optionally, Enterprise Manager can be configured to send an email message to the administrator about problem conditions as well as display alert information on the console.

You can also set thresholds on many of the pertinent metrics for your system. Oracle Database 11*g* proactively notifies you if the database deviates sufficiently from normal readings to reach those thresholds. An early notification of potential problems enables you to respond quickly and, in many cases, resolve issues before users even notice them.

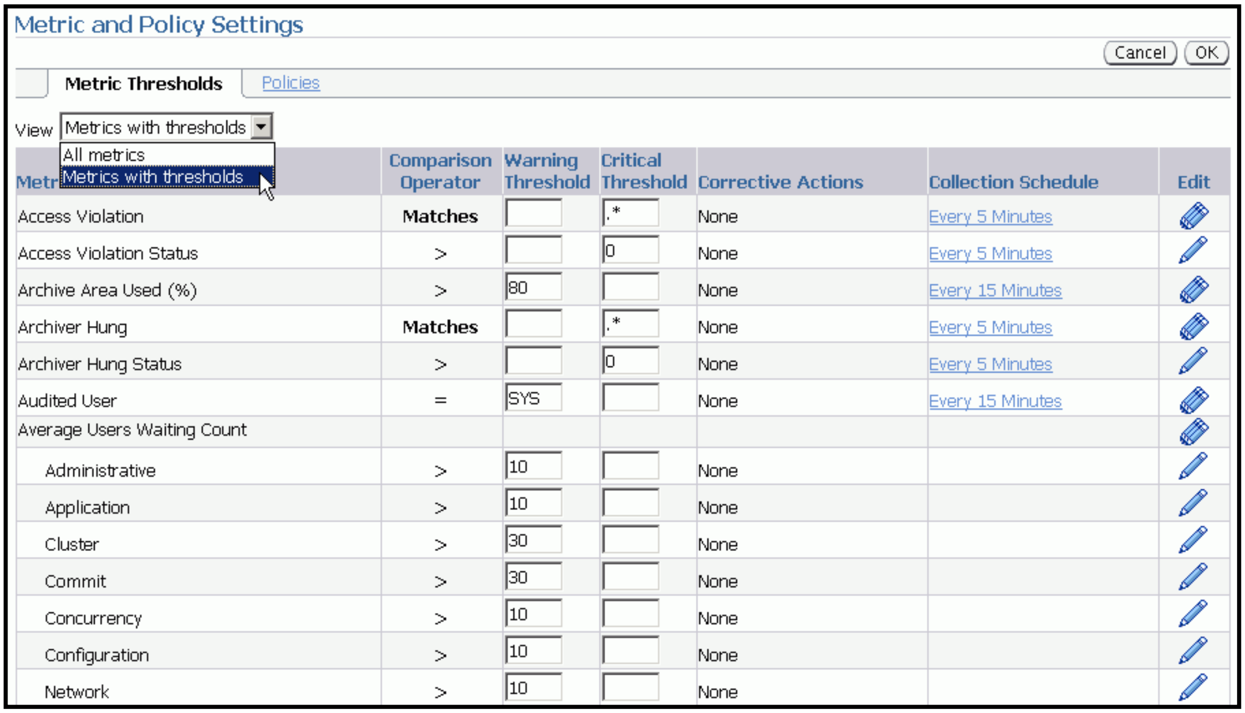
Approximately 60 metrics are monitored by default, among which are:

- Broken Job Count
- Database Time Spent Waiting (%)
- Dump Area Used (%)
- SQL Response Time (%) compared to baseline
- Tablespace Used (%)
- Generic Incident

A few additional key metrics can provide early problem notification:

- Average File Read Time (centiseconds)
- Response Time (per transaction)
- Wait Time (%)

Setting Thresholds



Metric and Policy Settings

View: Metrics with thresholds ▾ (All metrics / Metrics with thresholds)

Metr	Comparison Operator	Warning Threshold	Critical Threshold	Corrective Actions	Collection Schedule	Edit
Access Violation	**Matches**		.*	None	Every 5 Minutes	✎
Access Violation Status	>		0	None	Every 5 Minutes	✎
Archive Area Used (%)	>	80		None	Every 15 Minutes	✎
Archiver Hung	**Matches**		.*	None	Every 5 Minutes	✎
Archiver Hung Status	>		0	None	Every 5 Minutes	✎
Audited User	=	SYS		None	Every 15 Minutes	✎
Average Users Waiting Count						✎
Administrative	>	10		None		✎
Application	>	10		None		✎
Cluster	>	30		None		✎
Commit	>	30		None		✎
Concurrency	>	10		None		✎
Configuration	>	10		None		✎
Network	>	10		None		✎

Setting Thresholds

To set or edit a threshold for your whole database, click “Metric and Policy Settings” in the Related Links region of the database home page. Enter your desired warning and critical threshold values. The appropriate alerts appear when the database reaches your specified values.

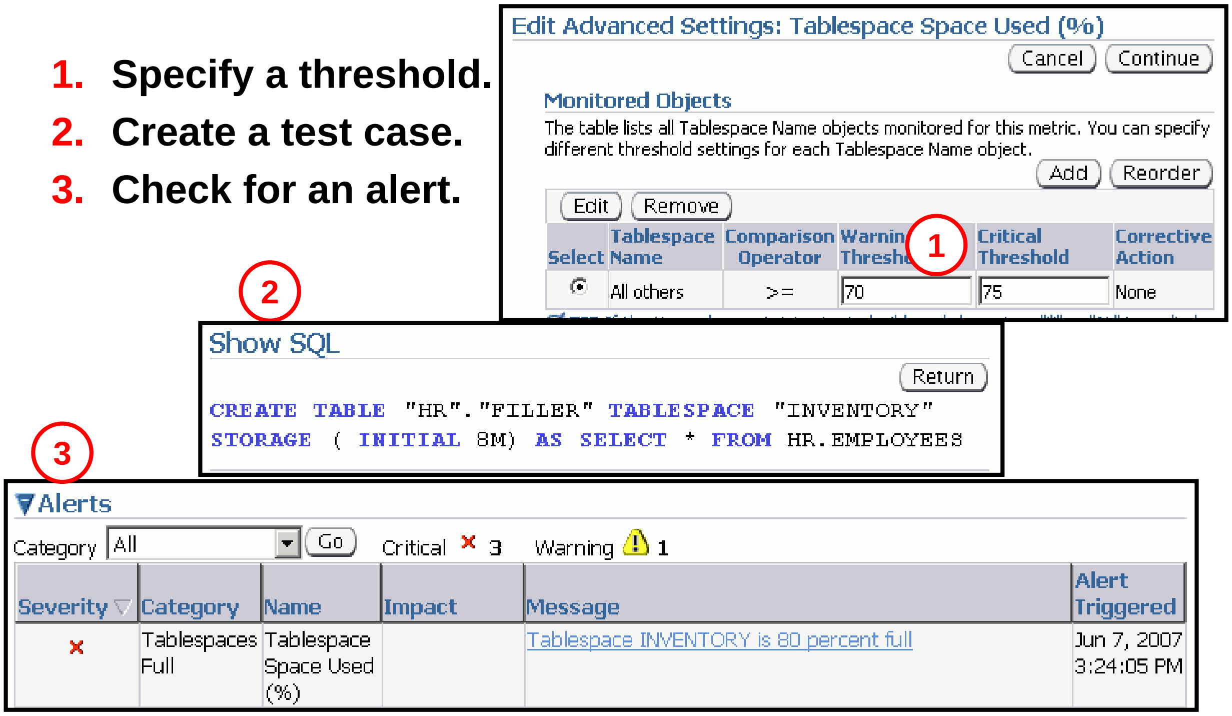
The thresholds that are already set appear in the “Metrics with thresholds” list. By default, approximately 60 metrics have preset thresholds; you may change these as needed. The “All metrics” list shows the metrics that do not have thresholds set.

Click one of the Edit icons to access a page where you can specify additional corrective actions for either warning or critical thresholds.

Click a Collection Schedule link and change the scheduled collection interval. Be aware that each schedule affects a group of metrics.

Creating and Testing an Alert

1. Specify a threshold.
2. Create a test case.
3. Check for an alert.



Edit Advanced Settings: Tablespace Space Used (%)

Monitored Objects

The table lists all Tablespace Name objects monitored for this metric. You can specify different threshold settings for each Tablespace Name object.

Select	Tablespace Name	Comparison Operator	Warnin Thresh	Critical Threshold	Corrective Action
●	All others	>=	70	75	None

Show SQL

```
CREATE TABLE "HR"."FILLER" TABLESPACE "INVENTORY"
STORAGE ( INITIAL 8M) AS SELECT * FROM HR.EMPLOYEES
```

Alerts

Category All Critical ✕ 3 Warning ⚠ 1

Severity	Category	Name	Impact	Message	Alert Triggered
✕	Tablespaces Full	Tablespace Space Used (%)		Tablespace INVENTORY is 80 percent full	Jun 7, 2007 3:24:05 PM

ORACLE

Creating and Testing an Alert

You can also set thresholds for a specific object.

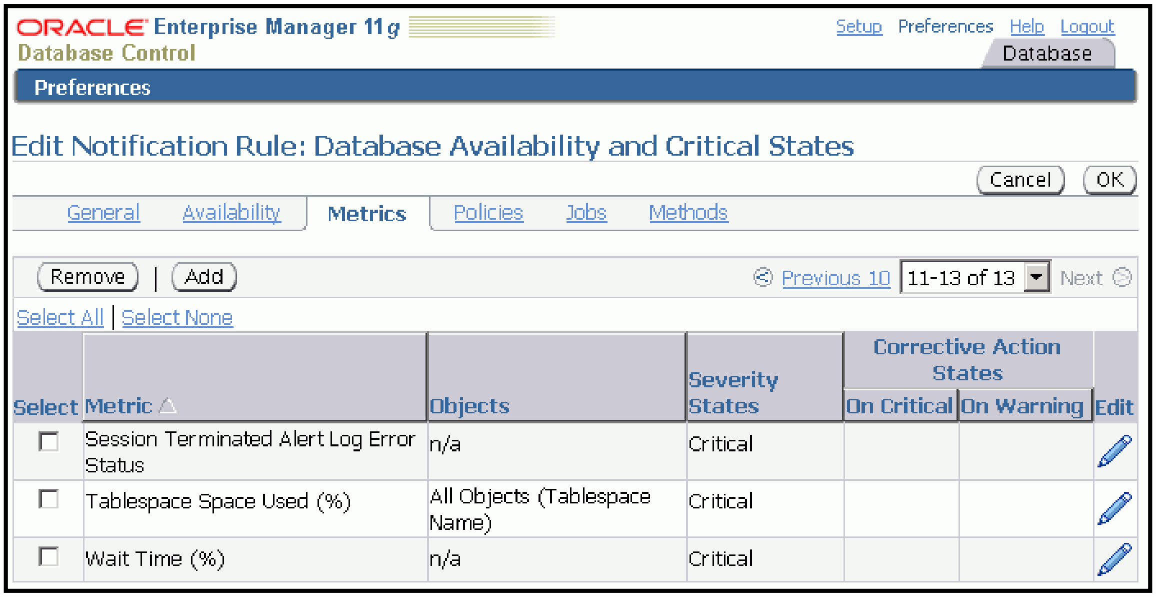
Example

You decide that you need to receive a critical alert if the space used in the `INVENTORY` tablespace exceeds 75%. (This tablespace does not allow its data files to automatically extend.) To create and test the alert, perform the following steps:

1. In Enterprise Manager, navigate to the “Metrics and Policy Settings” page, and then click the Edit icon for the Tablespace Used (%) threshold. Set your desired threshold for the tablespace.
2. Under the Schema tab on the Tables page, create a table to test the alert. Use the “Define using SQL” action to duplicate an existing table. The initial setting of 8 MB in the `STORAGE` clause causes the table to allocate 80% of the 10 MB `INVENTORY` tablespace immediately.
3. After you receive an error that this table is unable to extend, check the Database Home page for the associated alert. Tablespace Space Used (%) is collected every 10 minutes by default.

Most alerts contain the name of an associated advisor that can be invoked to give you more detailed advice. For each corresponding alert message, Database Control provides a link to invoke the corresponding advisor.

Alerts Notification



ORACLE Enterprise Manager 11*g*
Database Control

Setup Preferences Help Logout

Database

Preferences

Edit Notification Rule: Database Availability and Critical States

Cancel OK

Select	Metric	Objects	Severity States	Corrective Action States		Edit
				On Critical	On Warning	
☐	Session Terminated Alert Log Error Status	n/a	Critical			✎
☐	Tablespace Space Used (%)	All Objects (Tablespace Name)	Critical			✎
☐	Wait Time (%)	n/a	Critical			✎

Alerts Notification

The notification mechanism uses the Enterprise Manager user interface. It is based on the concept of a notification rule that establishes the appropriate notification mechanism for a set of upcoming alerts.

Using Database Control, you edit the notification rules. On the home page, click the Preferences link to display the General page, where you specify the email address at which you want to receive notifications.

On the General page, click the Rules link in the Notification region. Select the “Database Availability and Critical States” rule, and then click the Edit button. This takes you to the “Edit Notification Rule Database Availability and Critical States” page, where you click the Metrics tab and edit the metrics for which you want to receive notifications.

Alerts Notification (continued)

As an option, you can specify that Enterprise Manager provide you with direct notification when specific alerts arise. For example, if you specify that you want email notification for critical alerts, and you have a critical threshold set for the system response time for each call metric, you can send an email message containing a message similar to the following:

```
Host Name=mydb.us.mycompany.com
Metric=Response Time per Call
Timestamp=08-NOV-2005 10:10:01 (GMT -7:00)
Severity=Critical
Message=Response time per call has exceeded the threshold. See
  the latest ADDM analysis.
Rule Name= Rule
Owner=SYSMAN
```

The email contains a link to the host name and the latest ADDM analysis.

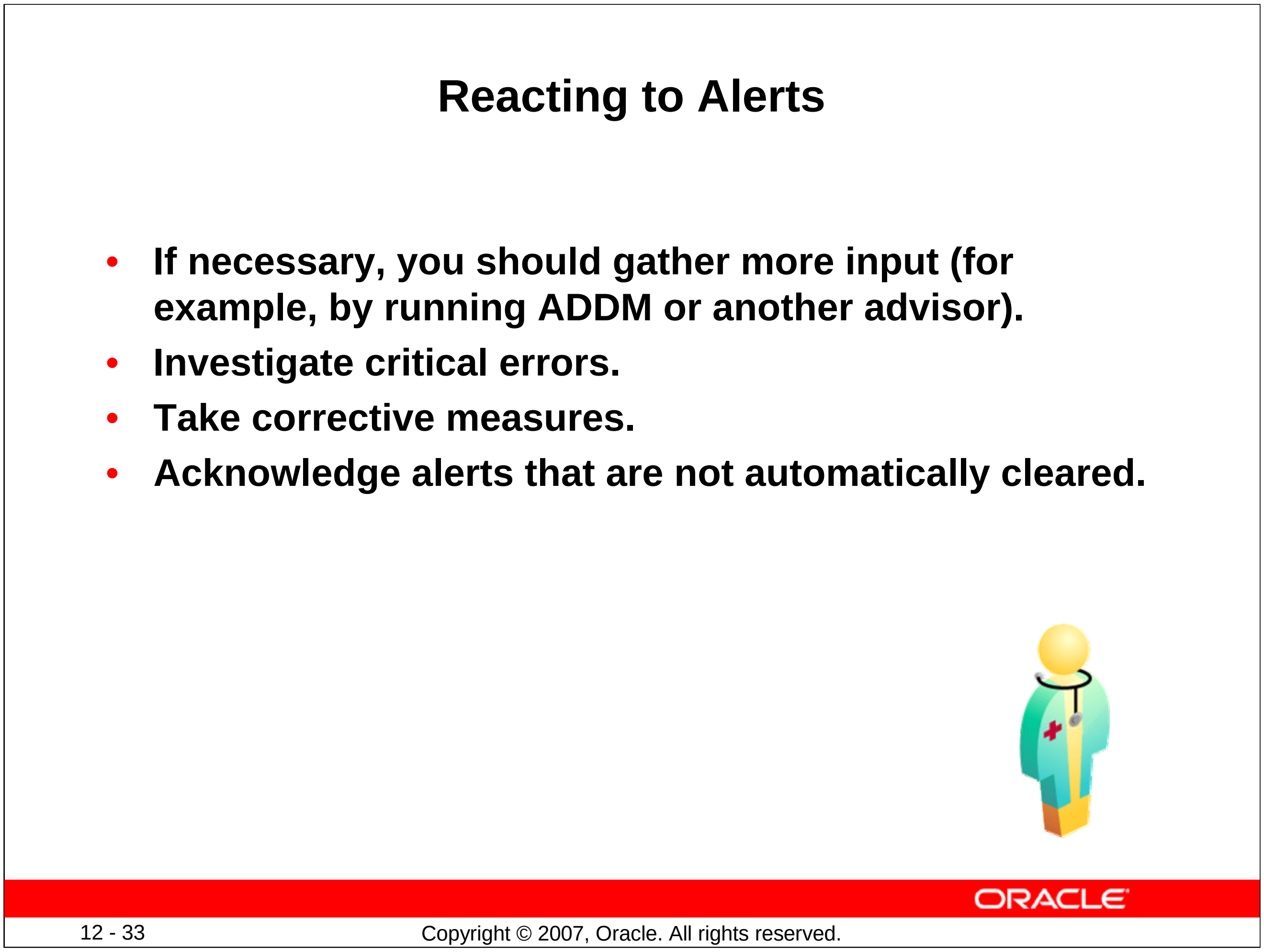
By default, alerts in critical state (such as DB Down, Generic Alert Log Error Status, and Tablespace Used) are set up for notification. However, to receive these notifications, you must set up your email information by following these steps:

1. On any Database Control page, click the Setup link in the header and footer area.
2. On the Setup page, select Notification Methods.
3. Enter the required information in the Mail Server region of the Notifications Methods page.

There are other methods of notification, including scripts and Simplified Network Management Protocol (SNMP) traps. The latter can be used to communicate with third-party applications.

To receive notifications:

1. On any Database Control page, click the Preferences link in the header and footer area.
2. On the Preferences page, select General. Enter your email address in the E-mail Addresses region.
3. You can optionally edit notification rules (for example, to change the severity state for receiving notification). To do so, click Notification Rules. The Notification Rules page appears.
   Note: For more information about configuring notification rules, see the *Oracle Enterprise Manager Advanced Configuration* documentation.



Reacting to Alerts

- If necessary, you should gather more input (for example, by running ADDM or another advisor).
- Investigate critical errors.
- Take corrective measures.
- Acknowledge alerts that are not automatically cleared.

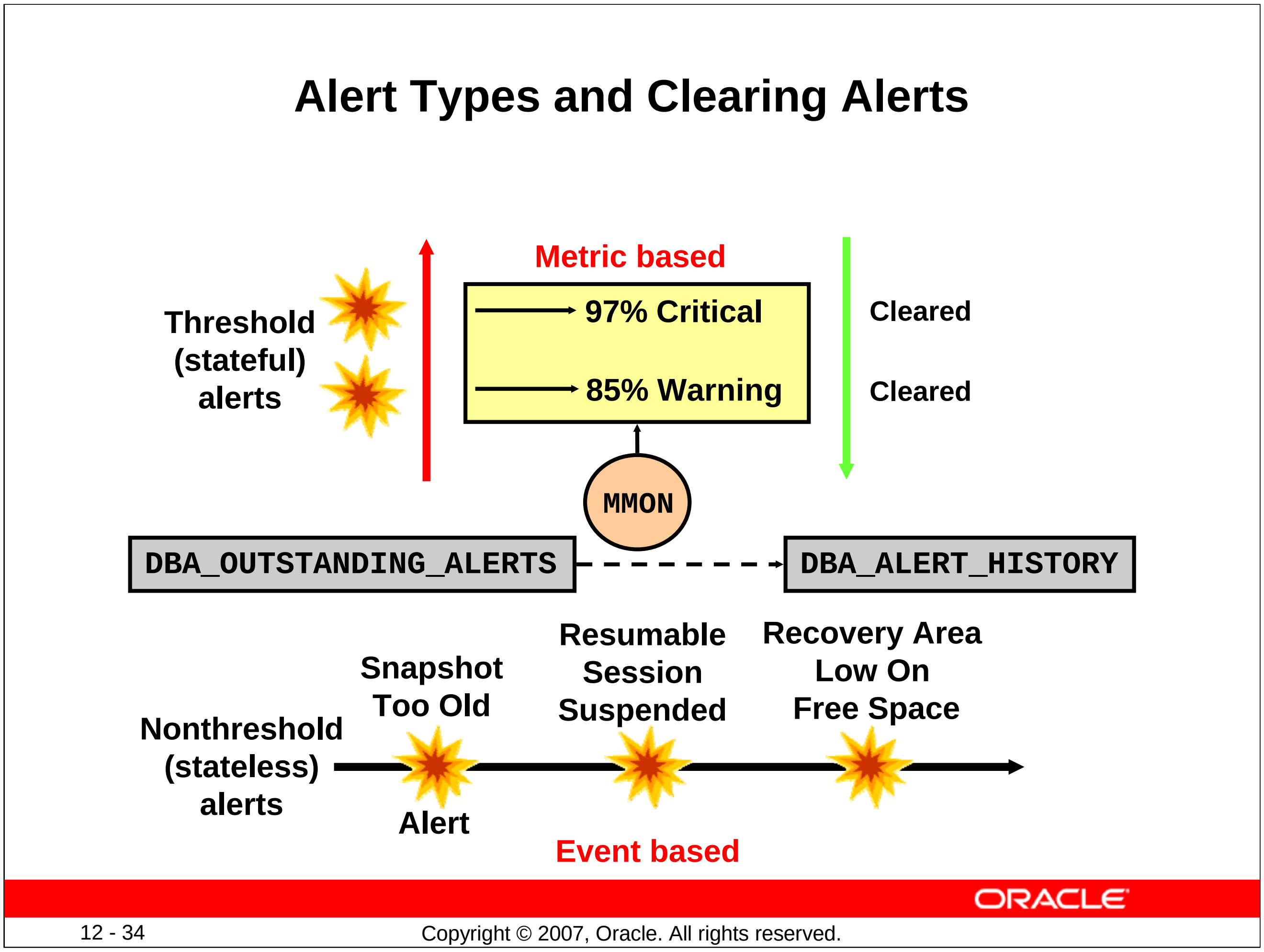
Reacting to Alerts

When you receive an alert, follow the recommendations that it provides. Or you can consider running the ADDM (or another advisor as appropriate) to obtain more detailed diagnostics of system or object behavior.

Alerts and incidents are generated for critical errors. Critical errors usually generate incidents that are collected into problems. You use the Support Workbench to investigate and possibly report the problem to Oracle Support.

Most alerts (such as “Out of Space”) are cleared automatically when the cause of the problem disappears. However, other alerts (such as Generic Alert Log Error) are sent to you for notification and must be acknowledged by you. After taking the necessary corrective measures, you acknowledge an alert by clearing or purging it. Clearing an alert sends the alert to the Alert History, which is viewable from the home page under Related Links. Purging an alert removes it from the Alert History.

To clear an alert such as Generic Alert Log Error, click the Alert Log link on the home page under Diagnostic Summary. The Alert Log Errors page appears. Select the alert to clear, and then click Clear. To purge an alert, select it and click Purge. You can also click the Clear Every Open Alert button or the Purge Every Alert button.



Alert Types and Clearing Alerts

Metric based

97% Critical

85% Warning

Threshold (stateful) alerts

Cleared

Cleared

MMON

DBA_OUTSTANDING_ALERTS

DBA_ALERT_HISTORY

Nonthreshold (stateless) alerts

Snapshot Too Old

Resumable Session Suspended

Recovery Area Low On Free Space

Alert

Event based

ORACLE

Alert Types and Clearing Alerts

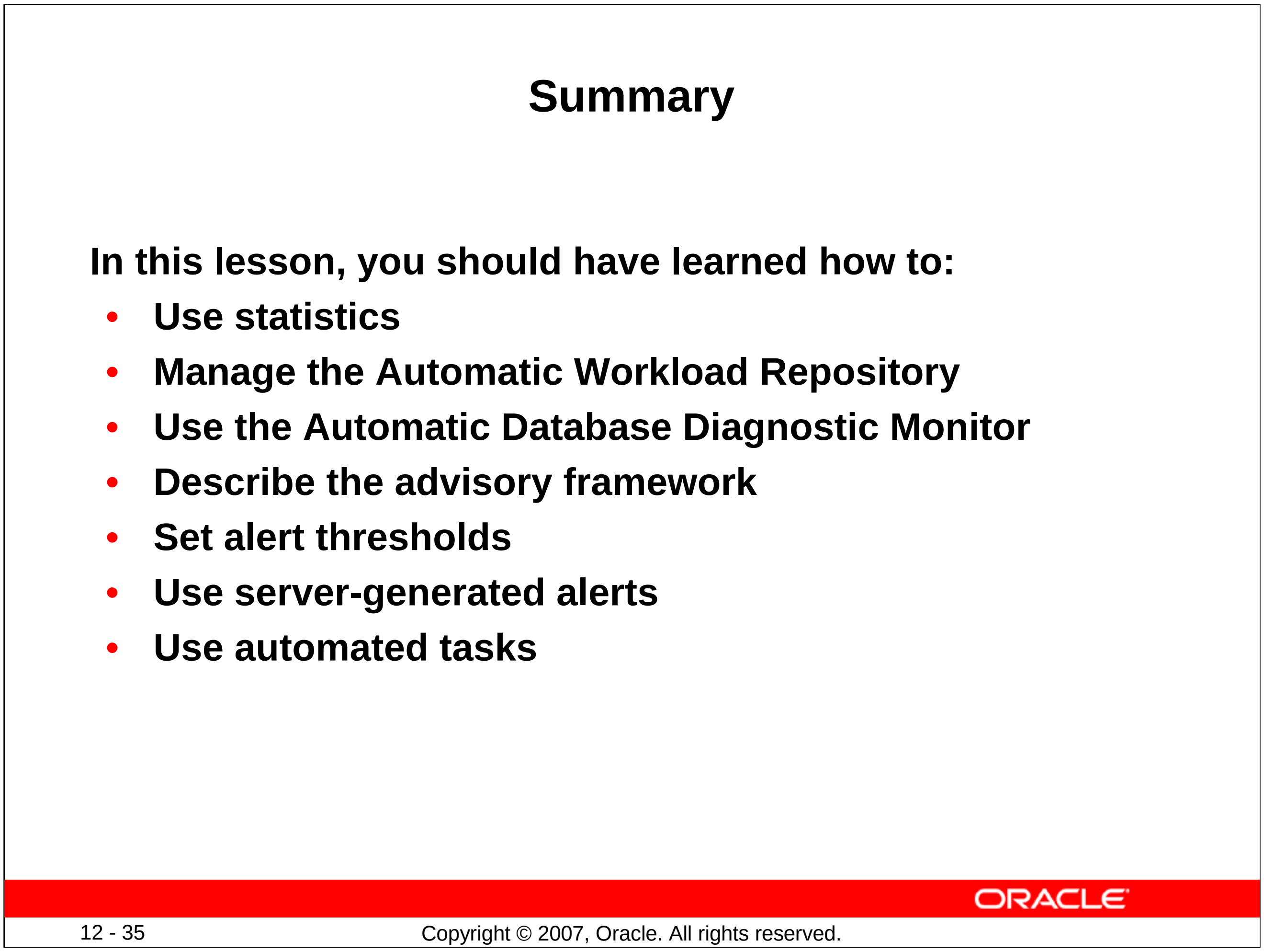
There are two kinds of server-generated alerts: threshold and nonthreshold.

Most server-generated alerts are configured by setting a warning and critical threshold values on database metrics. You can define thresholds for more than 120 metrics, including the following:

- Physical Reads Per Sec
- User Commits Per Sec
- SQL Service Response Time

Except for the Tablespace Space Usage metric, which is database related, the other metrics are instance related. Threshold alerts are also referred to as *stateful alerts*, which are automatically cleared when an alert condition clears. Stateful alerts appear in `DBA_OUTSTANDING_ALERTS` and, when cleared, go to `DBA_ALERT_HISTORY`.

Other server-generated alerts correspond to specific database events such as `ORA-*` errors, “Snapshot too old” errors, Recovery Area Low On Free Space, and Resumable Session Suspended. These are non-threshold-based alerts, also referred to as *stateless alerts*. Stateless alerts go directly to the history table. Clearing a stateless alert makes sense only in the Database Control environment because Database Control stores stateless alerts in its own repository.

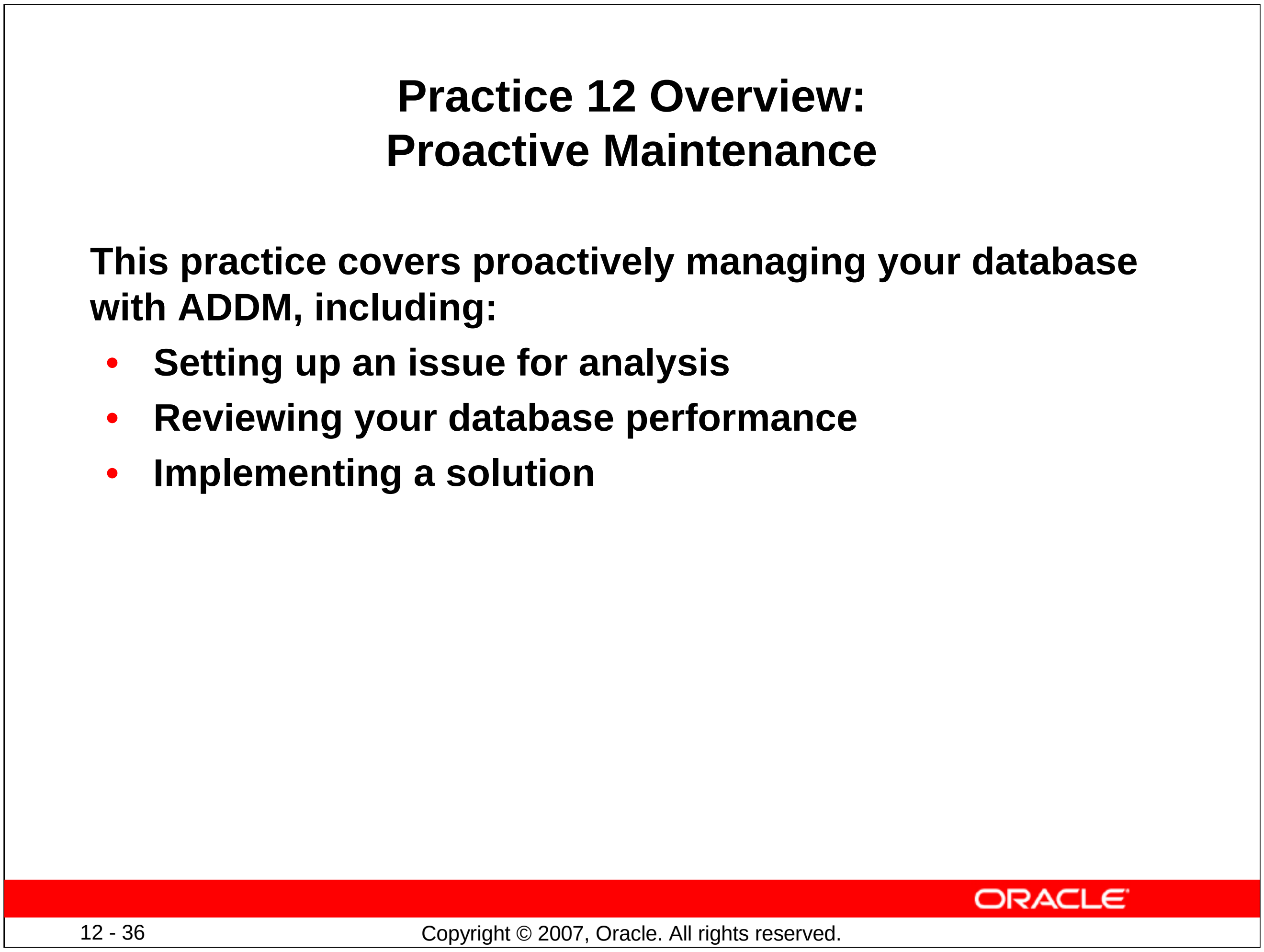


Summary

In this lesson, you should have learned how to:

- Use statistics
- Manage the Automatic Workload Repository
- Use the Automatic Database Diagnostic Monitor
- Describe the advisory framework
- Set alert thresholds
- Use server-generated alerts
- Use automated tasks

12 - 35



Practice 12 Overview: Proactive Maintenance

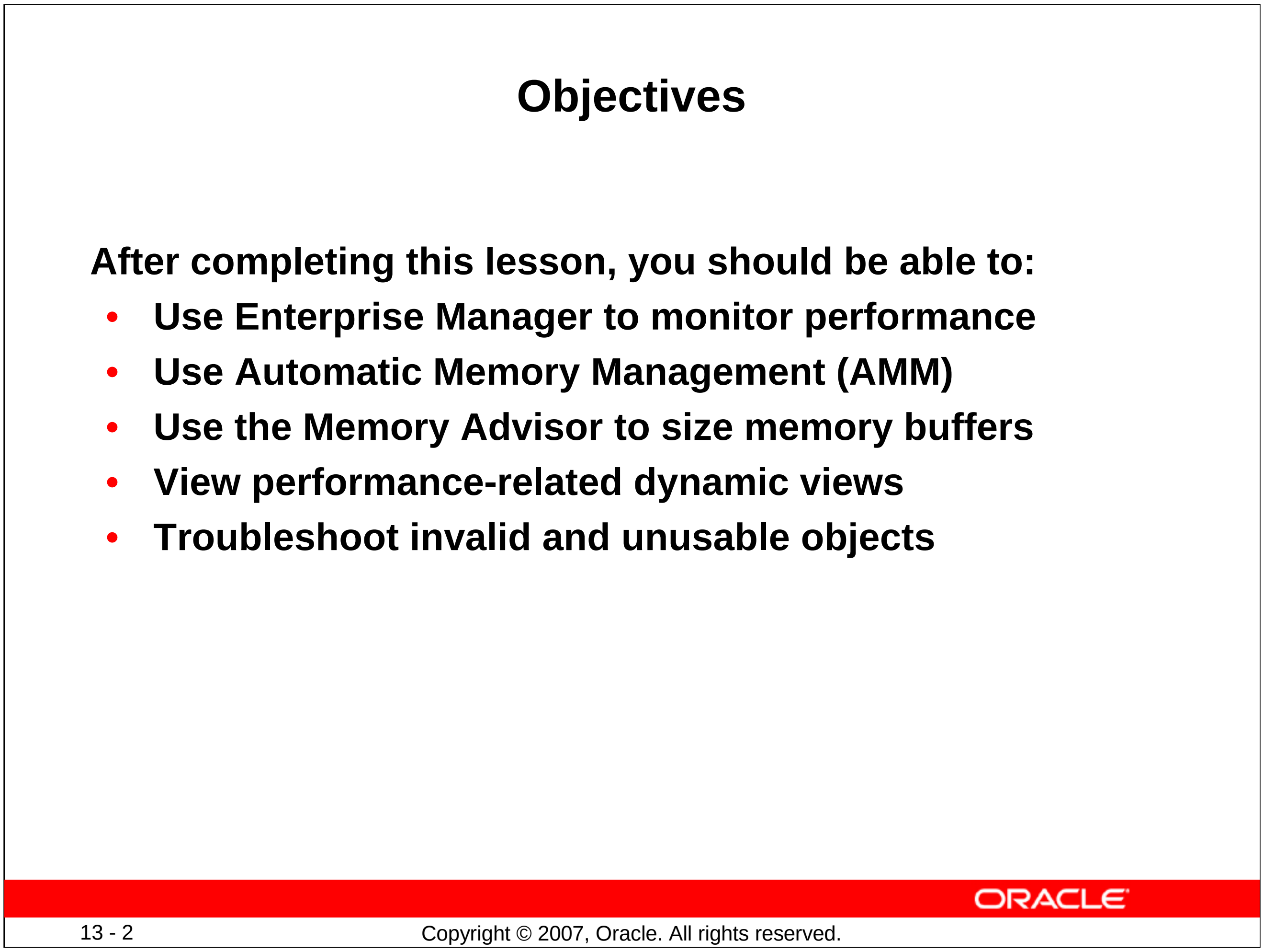
This practice covers proactively managing your database with ADDM, including:

- Setting up an issue for analysis
- Reviewing your database performance
- Implementing a solution

13

Performance Management

ORACLE

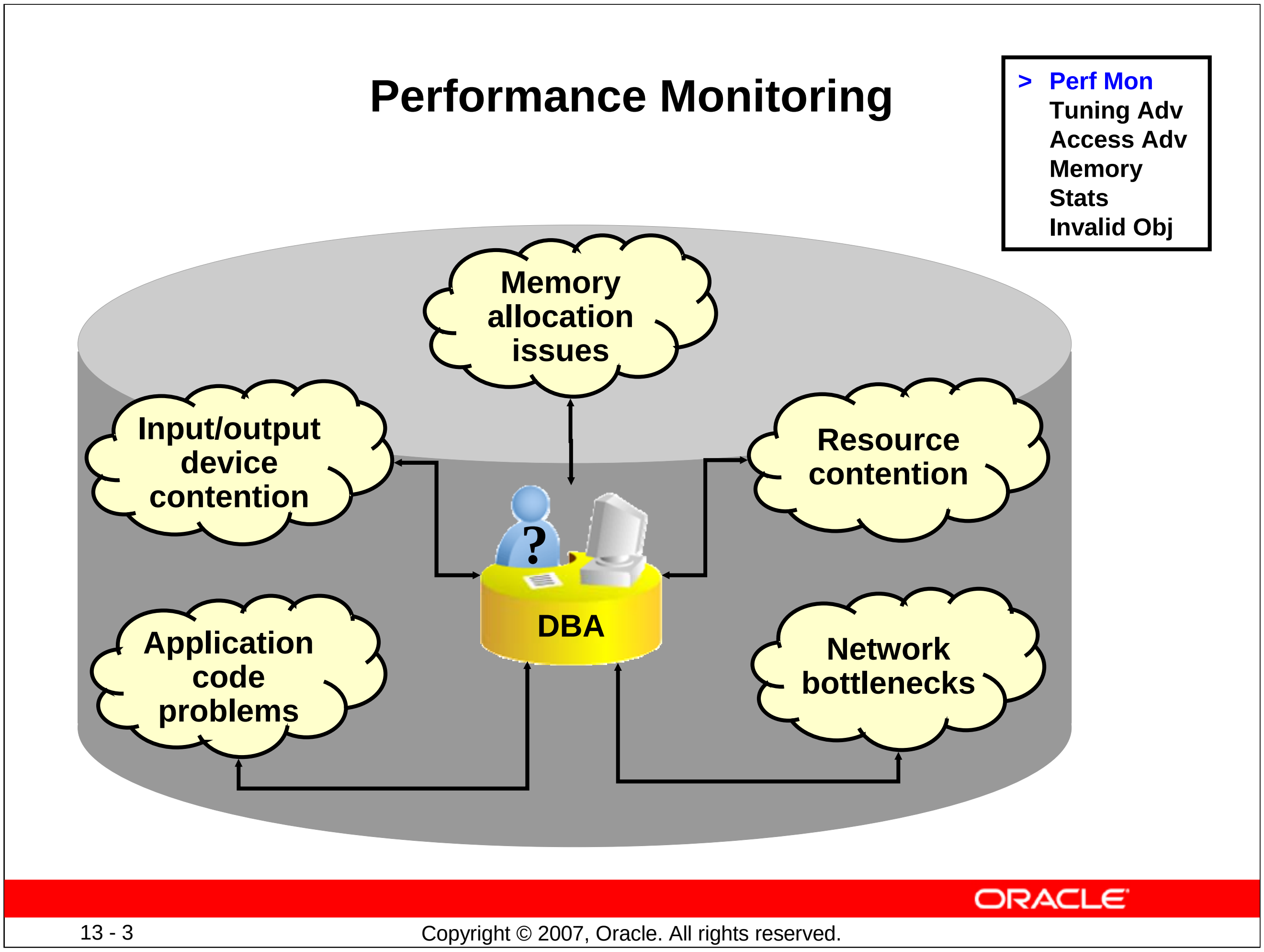


Objectives

After completing this lesson, you should be able to:

- Use Enterprise Manager to monitor performance
- Use Automatic Memory Management (AMM)
- Use the Memory Advisor to size memory buffers
- View performance-related dynamic views
- Troubleshoot invalid and unusable objects

13 - 2



Performance Monitoring

> Perf Mon
Tuning Adv
Access Adv
Memory
Stats
Invalid Obj

Memory allocation issues

Input/output device contention

Resource contention

DBA

Application code problems

Network bottlenecks

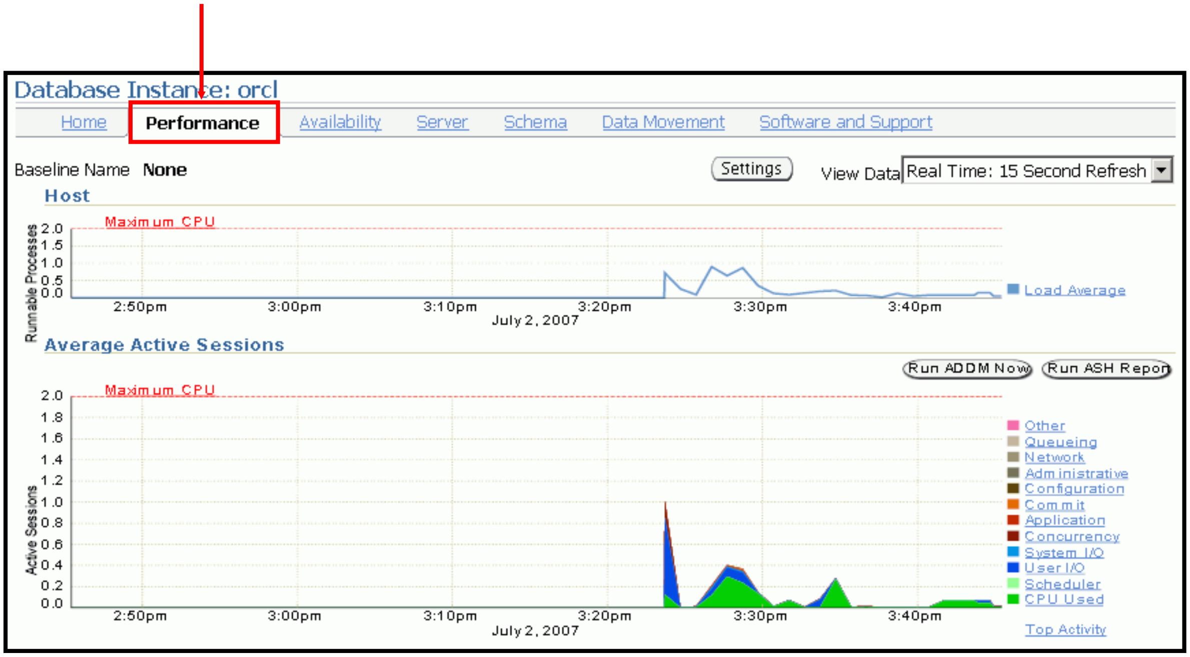
Performance Monitoring

To administer Oracle Database 11*g* and keep it running smoothly, the database administrator (DBA) must regularly monitor its performance to locate bottlenecks and correct problem areas.

A DBA can look at hundreds of performance measurements, covering everything from network performance and disk input/output (I/O) speed to the time spent working on individual application operations. These performance measurements are commonly referred to as *database metrics*.

Note: For more information about Oracle database performance, see the *Oracle Database 11g: Performance Tuning* course.

Enterprise Manager Performance Page



Database Instance: orcl

Baseline Name None

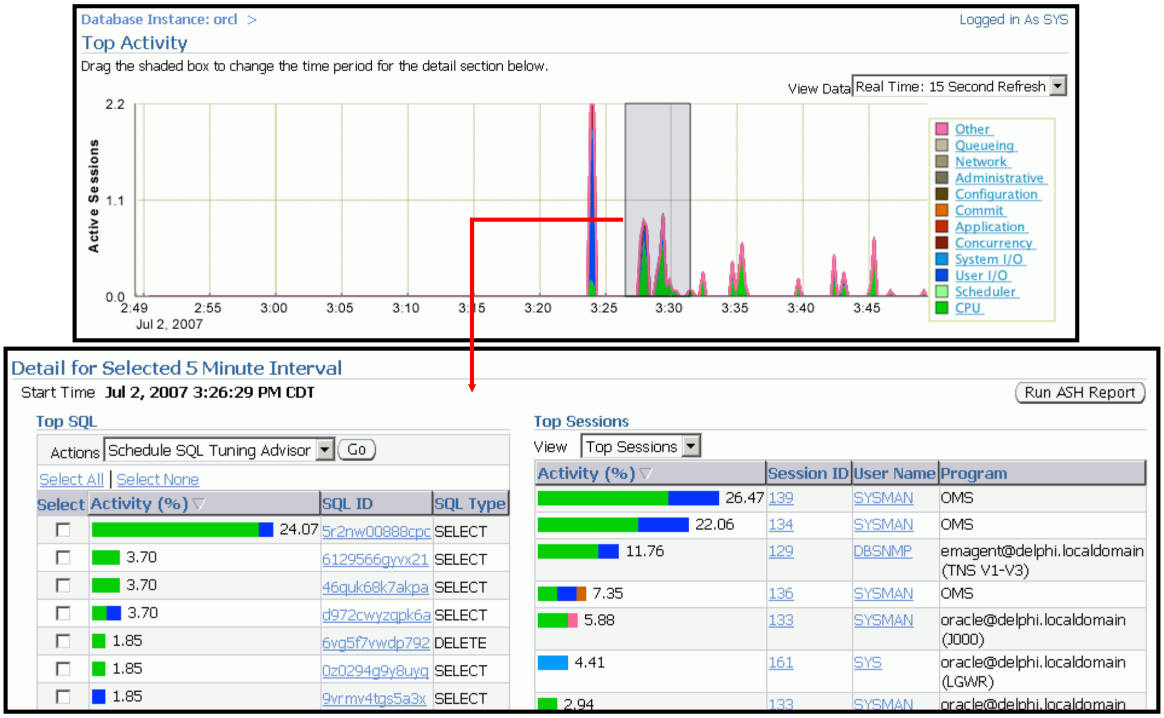
View Data Real Time: 15 Second Refresh

Enterprise Manager Performance Page

The Performance page in Enterprise Manager is the portal to a powerful set of performance monitoring and tuning tools. The first set of graphs on this page summarizes processes and active session activity. The Average Active Sessions graph shows the level of CPU usage and the resources that are causing the most wait events.

In the slide, you see that there was a recent increase in CPU usage and waits for User I/O, System I/O, and Concurrency. You can click these categories to see more details about the waits. The I/O data is divided into types of I/O (for example, log file read, control file write, and so on).

Active Session Page



Top Activity

Drag the shaded box to change the time period for the detail section below.

Detail for Selected 5 Minute Interval

Start Time **Jul 2, 2007 3:26:29 PM CDT**

Top SQL

Actions Schedule SQL Tuning Advisor

Select All | Select None

Select	Activity (%)	SQL ID	SQL Type
□	24.07	5r2nw00888cpc	SELECT
□	3.70	6129566gyvx21	SELECT
□	3.70	46quk68k7akpa	SELECT
□	3.70	d972cwyzqpk6a	SELECT
□	1.85	6vg5f7vwdp792	DELETE
□	1.85	0z0294g9y8uyq	SELECT
□	1.85	9vrmv4tgs5a3x	SELECT

Top Sessions

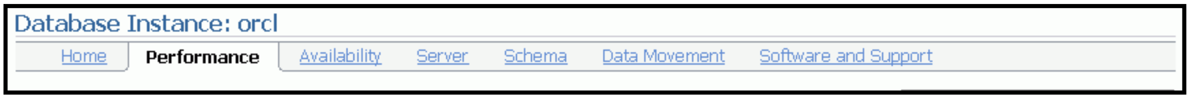
View Top Sessions

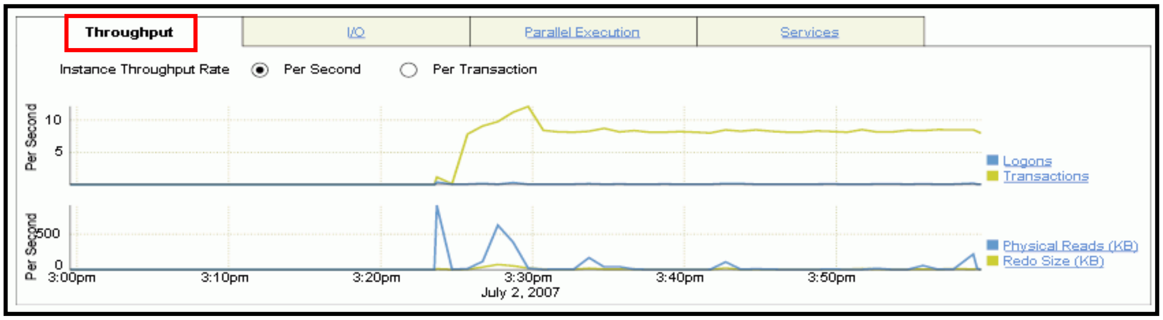
Activity (%)	Session ID	User Name	Program
26.47	139	SYSMAN	OMS
22.06	134	SYSMAN	OMS
11.76	129	DBSNMP	emagent@delphi.localdomain (TNS V1-V3)
7.35	136	SYSMAN	OMS
5.88	133	SYSMAN	oracle@delphi.localdomain (J000)
4.41	161	SYS	oracle@delphi.localdomain (LGWR)
2.94	133	SYSMAN	oracle@delphi.localdomain

Enterprise Manager Performance Page (continued)

When you drill down to a particular wait category, you can view details of specific five-minute intervals and also see the Top Working SQL and the Top Working Sessions associated with that particular wait event during that time. This enables you to perform after-the-fact analysis of system slowdowns and determine potential causes.

Performance Page: Throughput



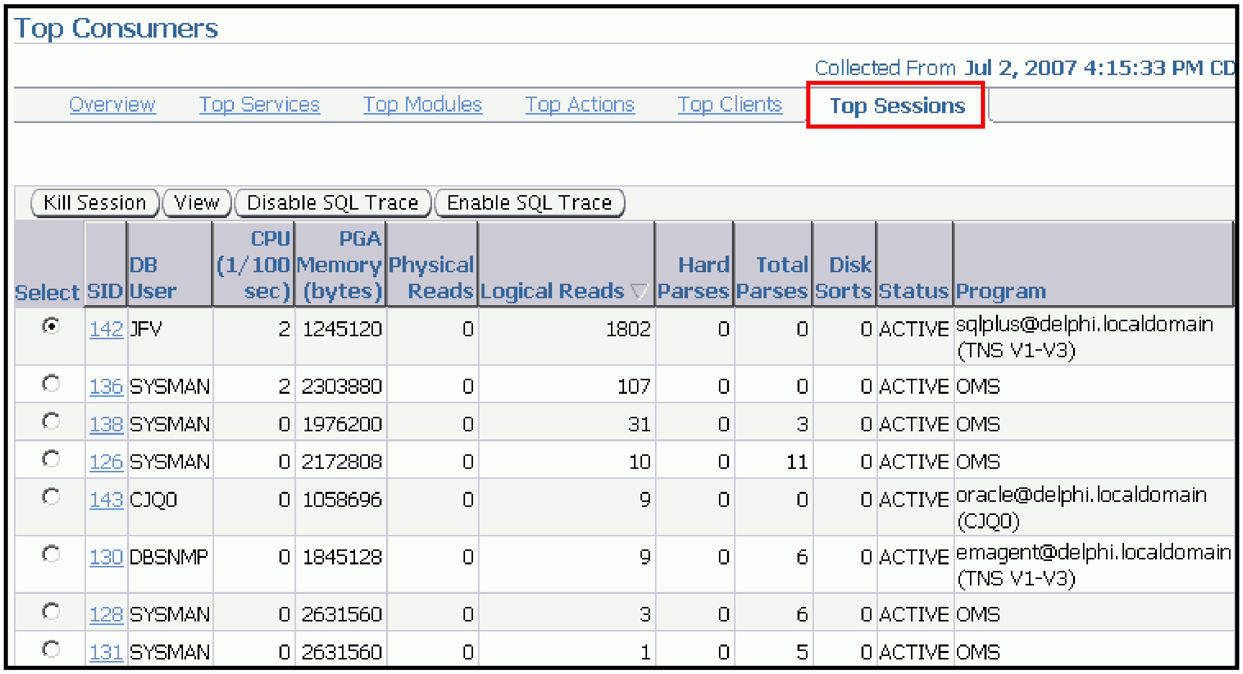


Instance Throughput Rate ● Per Second ○ Per Transaction

Performance Page: Throughput

You can see graphs of Instance throughput and Instance Disk I/O by clicking the Throughput and I/O tabs on the main Performance page. The Throughput tab has been selected in the slide.

Performance Monitoring: Top Sessions



Collected From **Jul 2, 2007 4:15:33 PM CD**

Select	SID	DB User	CPU (1/100 sec)	PGA Memory (bytes)	Physical Reads	Logical Reads ▽	Hard Parses	Total Parses	Disk Sorts	Status	Program
●	142	JFV	2	1245120	0	1802	0	0	0	ACTIVE	sqlplus@delphi.localdomain (TNS V1-V3)
○	136	SYSMAN	2	2303880	0	107	0	0	0	ACTIVE	OMS
○	138	SYSMAN	0	1976200	0	31	0	3	0	ACTIVE	OMS
○	126	SYSMAN	0	2172808	0	10	0	11	0	ACTIVE	OMS
○	143	CJQ0	0	1058696	0	9	0	0	0	ACTIVE	oracle@delphi.localdomain (CJQ0)
○	130	DBSNMP	0	1845128	0	9	0	6	0	ACTIVE	emagent@delphi.localdomain (TNS V1-V3)
○	128	SYSMAN	0	2631560	0	3	0	6	0	ACTIVE	OMS
○	131	SYSMAN	0	2631560	0	1	0	5	0	ACTIVE	OMS

Performance Monitoring: Top Sessions

Click Top Consumers in the Additional Monitoring Links section will take you to the Top Consumers page.

The Top Consumers Overview page shows in graphical format:

- Top services
- Top modules (by Service)
- Top Actions (by Service and Module)
- Top Clients

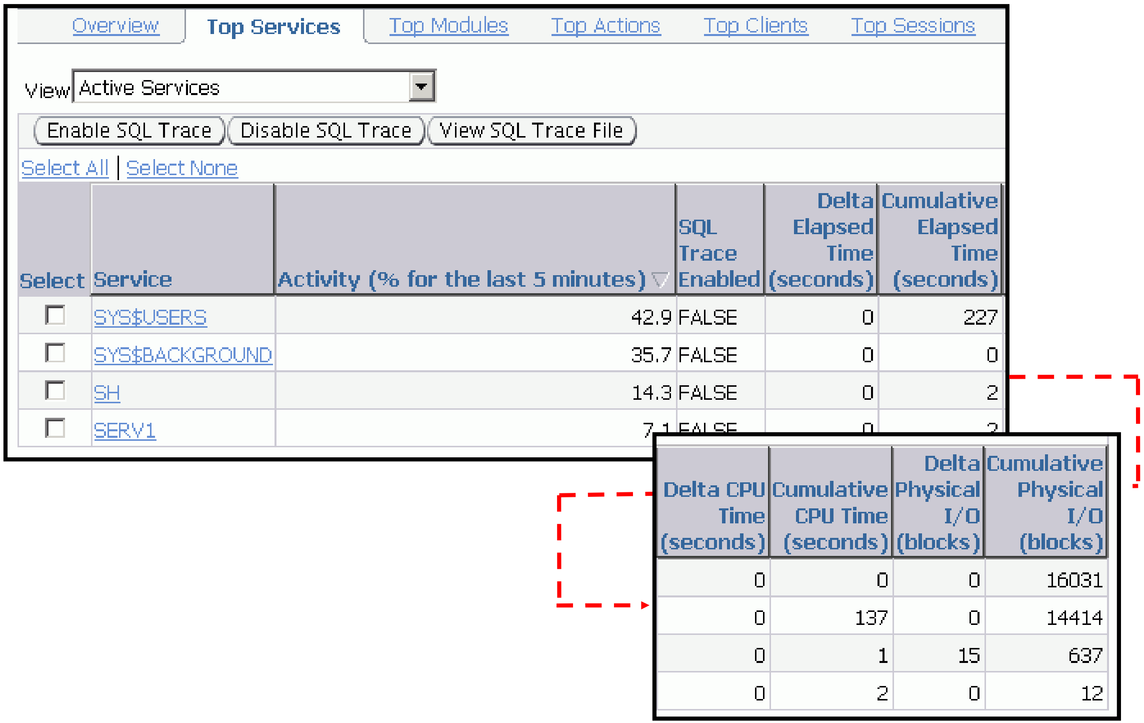
On the Top Consumers page, click the Top Sessions tab to see critical statistics of the sessions using the most resources

- CPU
- PGA Memory
- Logical Reads
- Physical Read
- Hard Parse count,
- Sort count.

If you click a column name, the associated statistic is the ordering value for the list.

The table on this page lists the sessions sorted by logical reads. This shows that the user JFV in session 142 is producing the greatest number of logical reads at this particular time.

Performance Monitoring: Top Services



View Active Services

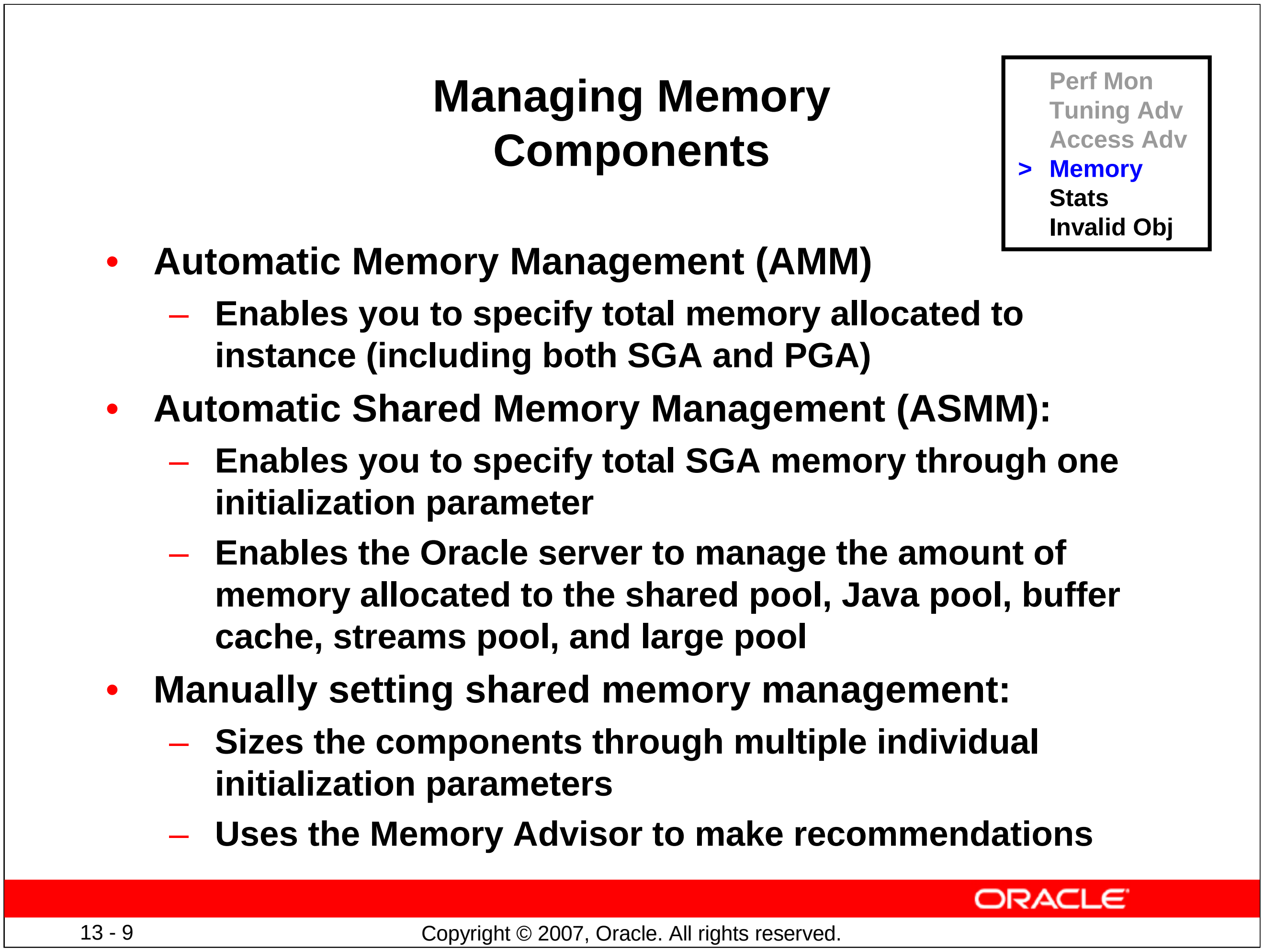
Select	Service	Activity (% for the last 5 minutes) ▽	SQL Trace Enabled	Delta Elapsed Time (seconds)	Cumulative Elapsed Time (seconds)
□	SYS$USERS	42.9	FALSE	0	227
□	SYS$BACKGROUND	35.7	FALSE	0	0
□	SH	14.3	FALSE	0	2
□	SERV1	7.1	FALSE	0	2

Delta CPU Time (seconds)	Cumulative CPU Time (seconds)	Delta Physical I/O (blocks)	Cumulative Physical I/O (blocks)
0	0	0	16031
0	137	0	14414
0	1	15	637
0	2	0	12

Performance Monitoring: Top Services

In multitier systems where there is an application server that is pooling database connections, viewing sessions may not provide the information you need to analyze performance. Grouping sessions into service names enables you to monitor performance more accurately.

In the example in the slide, there are three services: `inventory`, `orcl`, and `hr`. Regardless of the session that was used for a particular request, if it connected via one of these services, the performance data of the session is captured under that service name. Of the application services shown (SH and SERV1), it is clear from this listing that the SH service was more active during this five-minute interval.



Managing Memory Components

Perf Mon
Tuning Adv
Access Adv
> Memory
Stats
Invalid Obj

- Automatic Memory Management (AMM)
  - Enables you to specify total memory allocated to instance (including both SGA and PGA)
- Automatic Shared Memory Management (ASMM):
  - Enables you to specify total SGA memory through one initialization parameter
  - Enables the Oracle server to manage the amount of memory allocated to the shared pool, Java pool, buffer cache, streams pool, and large pool
- Manually setting shared memory management:
  - Sizes the components through multiple individual initialization parameters
  - Uses the Memory Advisor to make recommendations

13 - 9

Managing Memory Components

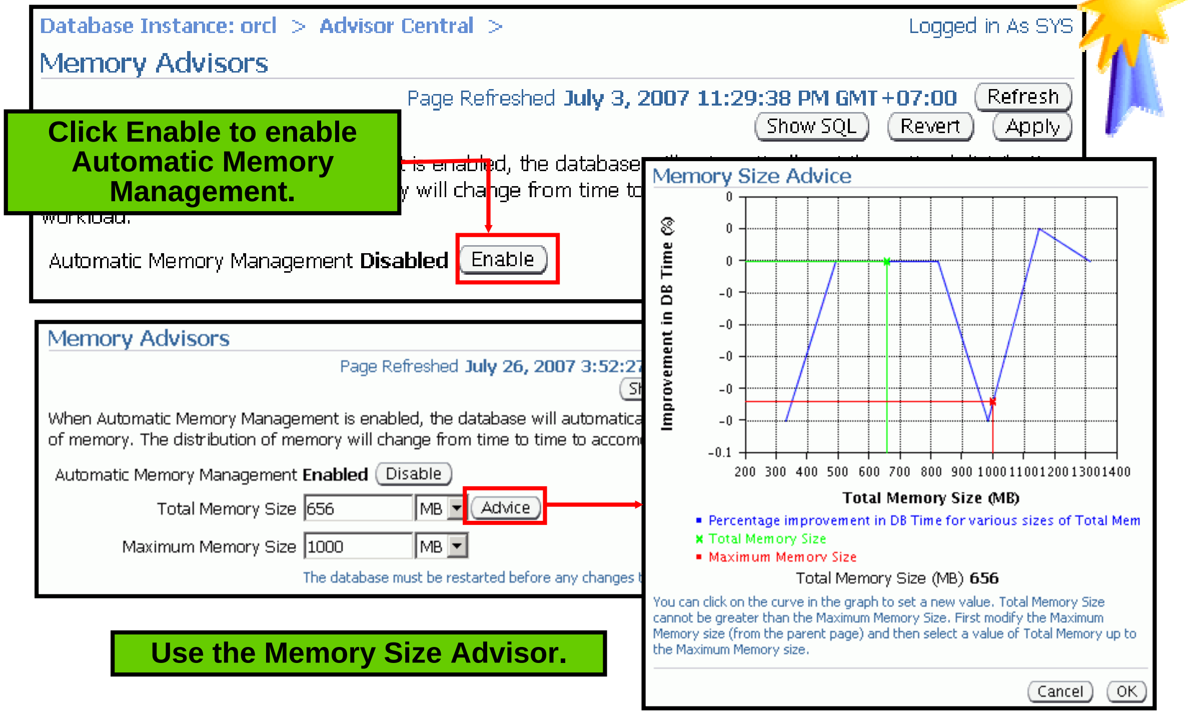
Oracle Database 11*g* enables you to specify the total memory allocated to the instance. Memory will be dynamically reallocated between the System Global Area (SGA) and Program Global Area (PGA) as needed. This method is called Automatic Memory Management (AMM) and is available on only those platforms that support dynamic release of memory. This simplifies your memory management tasks.

Memory advisors are available to help you set the initialization parameters on various levels. Which advisor is available depends on the level on which you are specifying the memory parameters. If you enable AMM, only the Memory Size Advisor is available.

Automatic Shared Memory Management (ASMM) enables you to manage the SGA as a whole. The SGA comprises several components. The sizes of many of these components are dynamically adjusted for best performance within the limits of the initialization parameters. When the AMM is enabled, the ASMM is automatically enabled. If the ASMM is enabled but not the AMM, the SGA Size Advisor is available.

You can manage the size of individual components manually by setting the initialization parameter for each component. If the Oracle server notifies you of a performance problem that is related to the size of an SGA or PGA component, you can use the Memory Advisor for the component to determine appropriate new settings. The Memory Advisor can model the effect of parameter changes.

Enabling Automatic Memory Management (AMM)



Memory Advisors

Page Refreshed **July 3, 2007 11:29:38 PM GMT+07:00**

t is enabled, the database
y will change from time to
workload.

Automatic Memory Management **Disabled**

Memory Advisors

Page Refreshed **July 26, 2007 3:52:27**

When Automatic Memory Management is enabled, the database will automatica
of memory. The distribution of memory will change from time to time to accom

Automatic Memory Management **Enabled**

Total Memory Size 656 MB

Maximum Memory Size 1000 MB

The database must be restarted before any changes t

Memory Size Advice

- Percentage improvement in DB Time for various sizes of Total Mem
- Total Memory Size
- Maximum Memory Size

Total Memory Size (MB) **656**

You can click on the curve in the graph to set a new value. Total Memory Size cannot be greater than the Maximum Memory Size. First modify the Maximum Memory size (from the parent page) and then select a value of Total Memory up to the Maximum Memory size.

Click Enable to enable Automatic Memory Management.

Use the Memory Size Advisor.

Enabling Automatic Memory Management (AMM)

If you did not enable Automatic Memory Management (AMM) when you configured your database, you can enable it by performing the following steps:

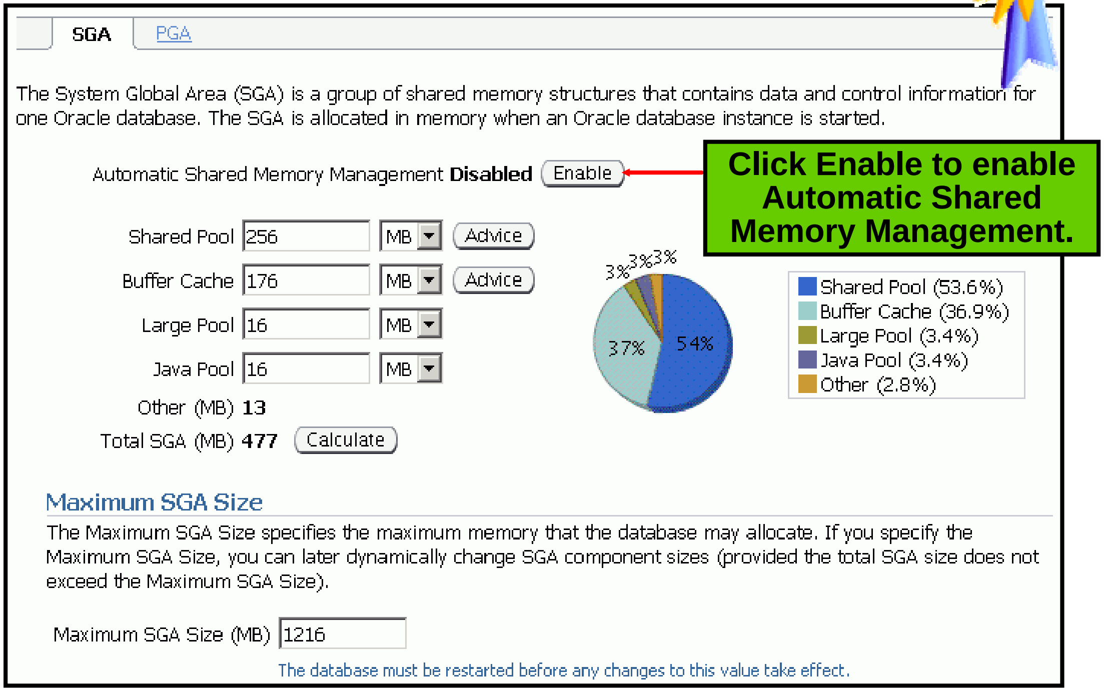
1. On the Database home page, click the Server tab.
2. Click Memory Advisors in the Database Configuration region.
   The Memory Advisors page appears.
3. Click Enable for Automatic Memory Management.
   The Enable Automatic Memory Management page appears.
4. Set the values for Total Memory Size and Maximum Memory Size for Automatic Memory Management.
   Note: If you change the Maximum Memory Size, the database instance must be restarted.
5. Click OK.

You can increase the size at a later time by increasing the value of the Total Memory Size field or the `MEMORY_TARGET` initialization parameter. However, you cannot set it higher than the value specified by the Maximum Memory Size field or the `MEMORY_MAX_TARGET` parameter. For more information, see the *Oracle Database Administrator's Guide*.

After AMM is enabled, the Memory Size Advisor is available to help you adjust the maximum and target memory sizes.

Note: Oracle recommends that you use Automatic Memory Management to simplify memory management tasks.

Enabling Automatic Shared Memory Management (ASMM)



SGA | PGA

The System Global Area (SGA) is a group of shared memory structures that contains data and control information for one Oracle database. The SGA is allocated in memory when an Oracle database instance is started.

Automatic Shared Memory Management **Disabled** Enable

Shared Pool 256 MB Advice

Buffer Cache 176 MB Advice

Large Pool 16 MB

Java Pool 16 MB

Other (MB) **13**

Total SGA (MB) **477** Calculate

🟦 Shared Pool (53.6%)
🟩 Buffer Cache (36.9%)
🟨 Large Pool (3.4%)
🟪 Java Pool (3.4%)
🟧 Other (2.8%)

Maximum SGA Size

The Maximum SGA Size specifies the maximum memory that the database may allocate. If you specify the Maximum SGA Size, you can later dynamically change SGA component sizes (provided the total SGA size does not exceed the Maximum SGA Size).

Maximum SGA Size (MB) 1216

The database must be restarted before any changes to this value take effect.

Click Enable to enable Automatic Shared Memory Management.

Enabling Automatic Shared Memory Management (ASMM)

Automatic Shared Memory Management is automatically enabled if you have enabled AMM. If you have not enabled AMM or did not enable ASMM when you configured your database, you can enable Automatic Shared Memory Management by performing the following steps:

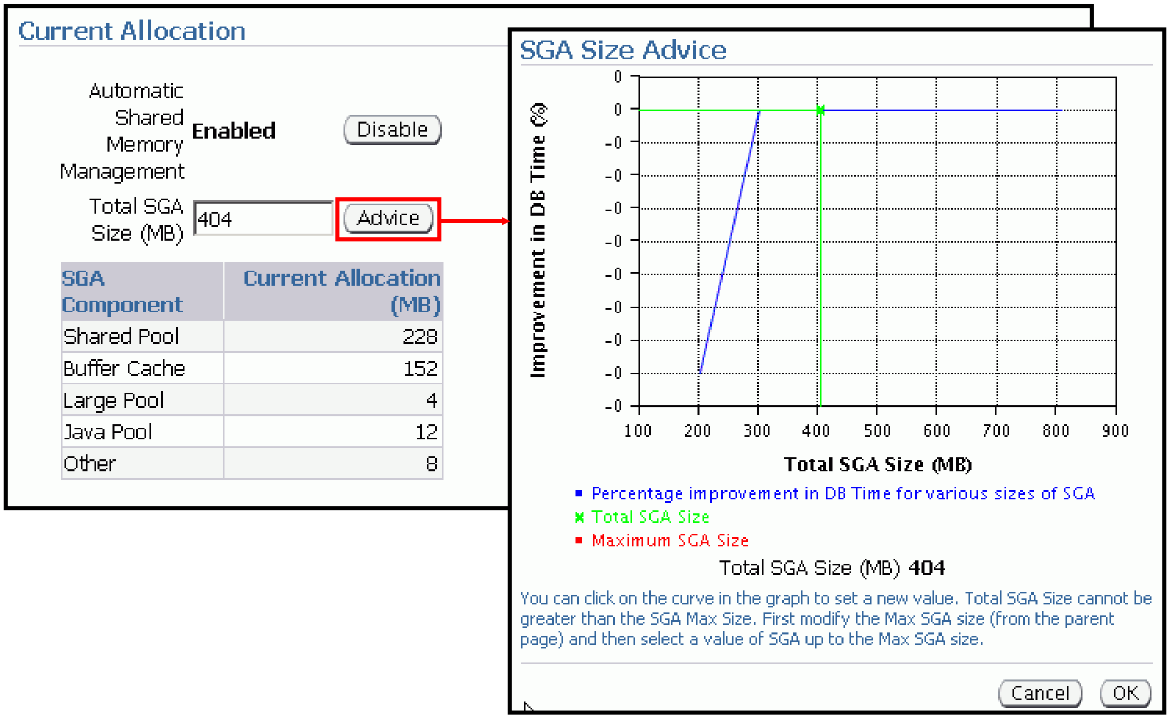
1. On the Database home page, click the Server tab.
2. Click Memory Advisors in the Database Configuration region.
   The Memory Advisors page appears.
3. Scroll down to the SGA section. Click Enable for Automatic Shared Memory Management.
   The Enable Automatic Shared Memory Management page appears.
4. Specify the total SGA size. Click OK.

You can increase the total SGA size at a later time by increasing the value of the Total SGA Size field or the `SGA_TARGET` initialization parameter. However, you cannot set it higher than the value specified by the Maximum SGA Size field or the `SGA_MAX_SIZE` parameter. For more information, see the *Oracle Database Administrator’s Guide*.

When AMM is disabled, the PGA advisor is accessible. The PGA advisor is recommended for setting the PGA memory value. Click the PGA tab to access the PGA property page. Click Advice to invoke the PGA Advisor.

Note: Oracle recommends that you use Automatic Shared Memory Management to simplify your memory management tasks.

Automatic Shared Memory Advisor



SGA Component	Current Allocation (MB)
Shared Pool	228
Buffer Cache	152
Large Pool	4
Java Pool	12
Other	8

- ■ Percentage improvement in DB Time for various sizes of SGA
- ✖ Total SGA Size
- ■ Maximum SGA Size

Total SGA Size (MB) **404**

You can click on the curve in the graph to set a new value. Total SGA Size cannot be greater than the SGA Max Size. First modify the Max SGA size (from the parent page) and then select a value of SGA up to the Max SGA size.

Automatic Shared Memory Advisor

When ASMM is enabled, you cannot set the initialization parameters for the specific components of shared memory that ASMM manages.After ASMM is enabled, the SGA Size Advisor is available to help you choose the best value for total SGA size.

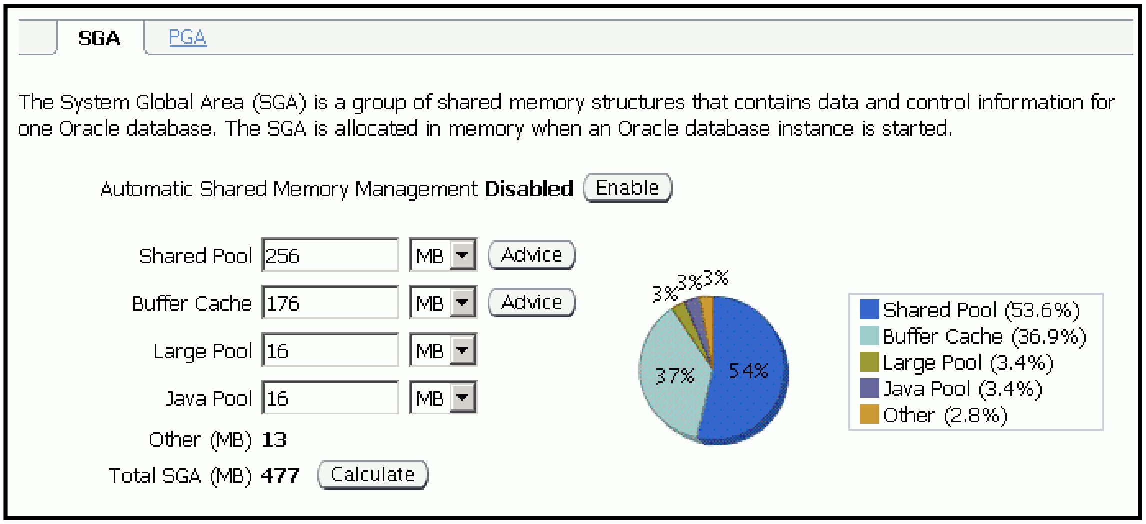
If ASMM is enabled, you should not initially set initialization parameters for the specific components for which it manages memory. If, after seeing the effects of the ASMM allocations, you decide that you want to adjust certain component allocations, you can specify values for those components. Those values are treated as minimum memory sizes for their respective components. Doing this limits the amount of memory available for automatic adjustment, but the capability is available if your environment requires special sizing that is not accommodated by ASMM.

The initialization parameters of concern are the following:

- `SHARED_POOL_SIZE`
- `LARGE_POOL_SIZE`
- `JAVA_POOL_SIZE`
- `DB_CACHE_SIZE`
- `STREAMS_POOL_SIZE`

To adjust these parameters while ASMM is enabled, you must use the `ALTER SYSTEM` command.

Setting Shared Memory Components Manually



SGA | PGA

The System Global Area (SGA) is a group of shared memory structures that contains data and control information for one Oracle database. The SGA is allocated in memory when an Oracle database instance is started.

Automatic Shared Memory Management **Disabled** Enable

Shared Pool	256	MB	Advice
Buffer Cache	176	MB	Advice
Large Pool	16	MB	
Java Pool	16	MB	
Other (MB)	**13**		
Total SGA (MB)	**477**	Calculate	

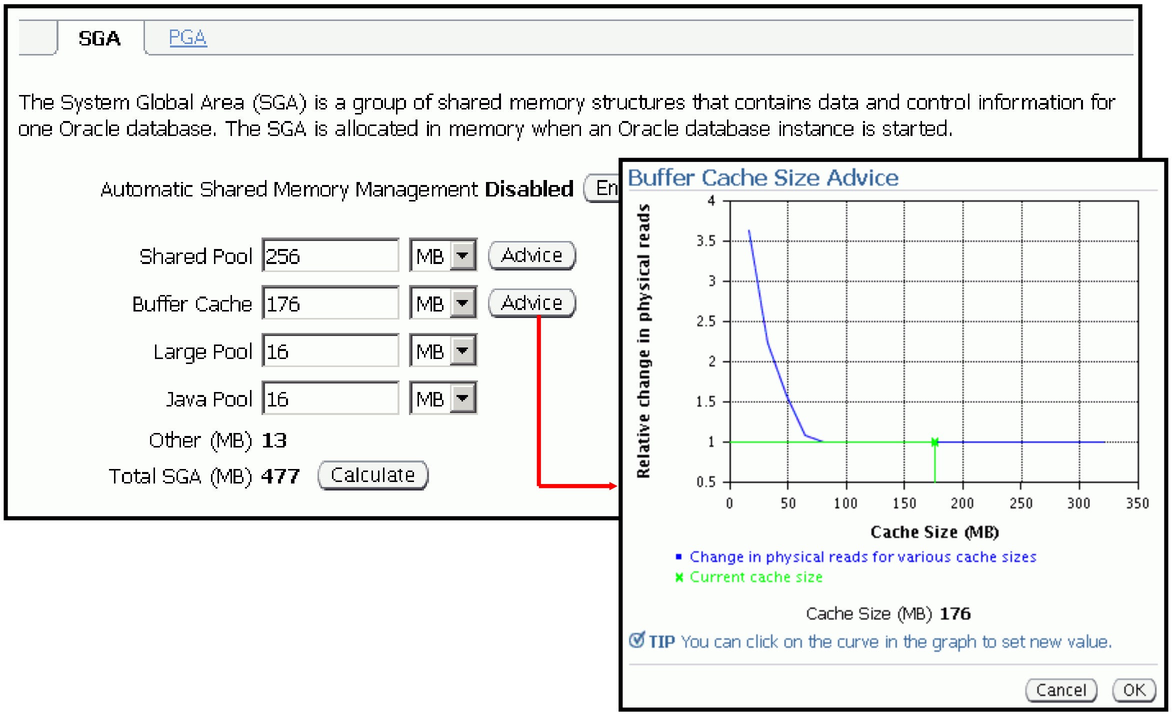
Setting Shared Memory Components Manually

If you do not use Automatic Shared Memory Management, you must provide values for each component of the SGA on installation and database creation.

To adjust the memory parameters:

1. Access the Memory Advisors page by clicking the Memory Advisors link in the Database Configuration region.
2. Invoke any of the advisors by clicking Advice beside the component specification. Click Help to view online help for additional information about how the advisor works.
3. In the component specification fields, enter new values based on advisor results or your own monitoring.

Using Memory Advisors



SGA | PGA

The System Global Area (SGA) is a group of shared memory structures that contains data and control information for one Oracle database. The SGA is allocated in memory when an Oracle database instance is started.

Automatic Shared Memory Management **Disabled**

Shared Pool 256 MB

Buffer Cache 176 MB

Large Pool 16 MB

Java Pool 16 MB

Other (MB) **13**

Total SGA (MB) **477**

Buffer Cache Size Advice

Cache Size (MB) **176**

TIP You can click on the curve in the graph to set new value.

Using Memory Advisors

The component-level memory advisors help you tune the size of your memory structures. You can use these advisors only when automatic memory management and automatic shared memory management are disabled.

The Memory Advisor comprises three advisors available from Enterprise Manager that give you recommendations on the following memory structures:

- Shared pool in the System Global Area (SGA)
- Buffer cache in the SGA

You can invoke the memory advisors by performing the following steps:

1. Click Advisor Central in the Related Links region on the Database home page.
2. Click Memory Advisor on the Advisor Central page.
   The Memory Advisors page provides a breakdown of memory usage for the SGA.
   Note: The Automatic Shared Memory Management setting must be disabled to run the individual component advisors.
3. Click Advice next to the Shared Pool value or Buffer Cache value to invoke the respective advisors.

Using Memory Advisors (continued)

All the parts of the Memory Advisor can be accessed through SQL*Plus by viewing the associated `V$*` views. There are four advisor views for the individual components of the SGA that are automatically tunable.

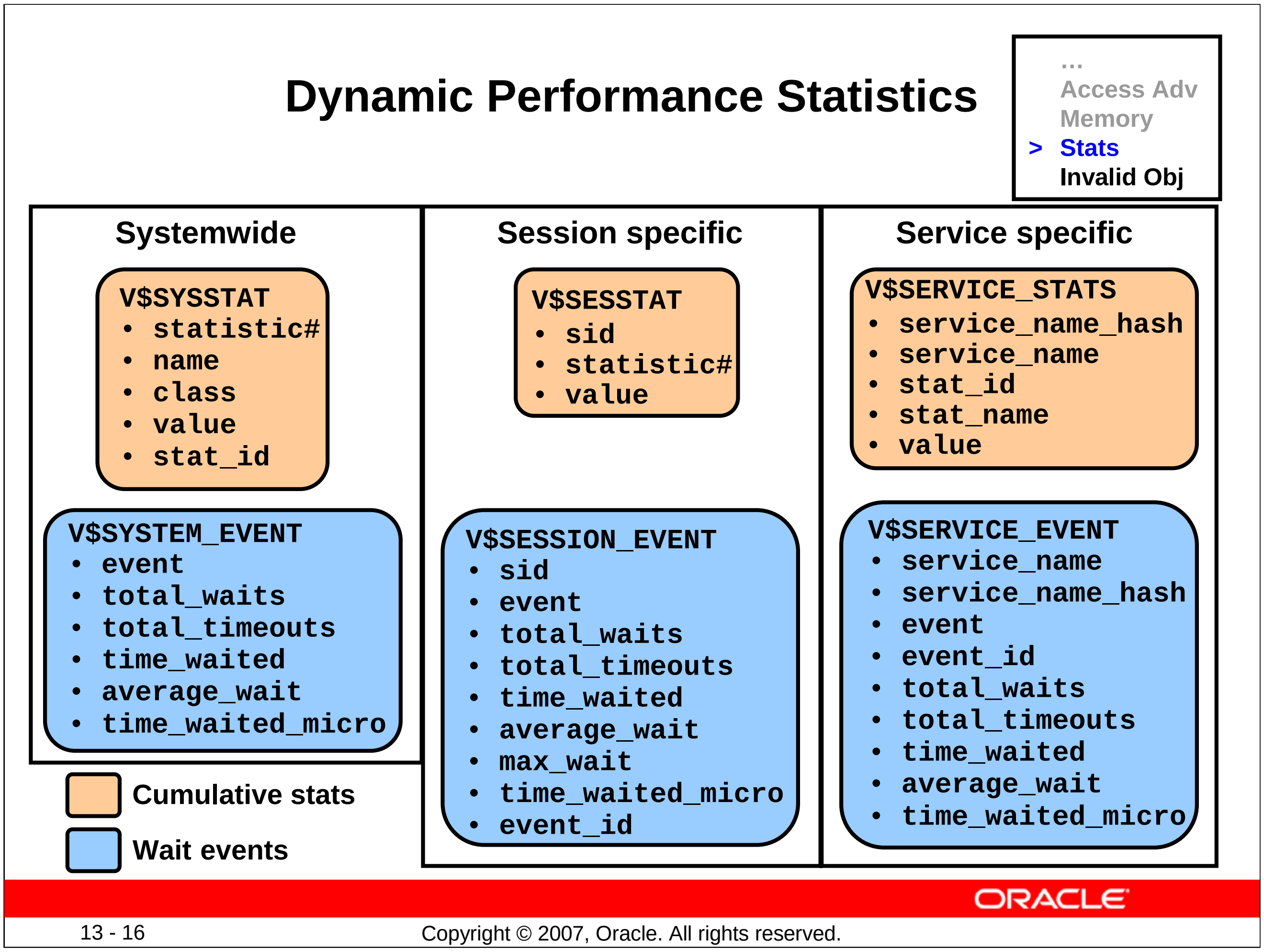
To help you size the most important SGA components, a number of component advisors have been introduced in the Oracle database. They are:

- **`V$DB_CACHE_ADVICE`:** Contains rows that predict the number of physical reads and time for the cache size corresponding to each row
- **`V$SHARED_POOL_ADVICE`:** Displays information about estimated parse time in the shared pool for different pool sizes
- **`V$JAVA_POOL_ADVICE`:** Displays information about estimated class load time into the Java pool for different pool sizes
- **`V$STREAMS_POOL_ADVICE`:** Displays information about the estimated count of spilled or unspilled messages and the associated time spent in the spill or unspill activity for different Streams pool sizes

Note: For more information about these views, see the *Oracle Database Reference.*

Sample SQL statement to access a memory advisor:

```
SQL> SELECT size_for_estimate, estd_physical_read_factor
   >  FROM V$DB_CACHE_ADVICE;

SIZE_FOR_ESTIMATE ESTD_PHYSICAL_READ_FACTOR
----------------- -------------------------
               16                    4.0296
               32                    2.4861
               48                    1.7561
               64                    1.0993
               80                    1.0017
               96                         1
              112                         1
              128                         1
              144                         1
              160                         1
              176                         1
              192                         1
              208                         1
              224                         1
              240                         1
              256                         1
              272                         1
              288                         1
              304                         1
              320                         1
```



Dynamic Performance Statistics

…
Access Adv
Memory
> Stats
Invalid Obj

Systemwide

V$SYSSTAT
- statistic#
- name
- class
- value
- stat_id

V$SYSTEM_EVENT
- event
- total_waits
- total_timeouts
- time_waited
- average_wait
- time_waited_micro

Session specific

V$SESSTAT
- sid
- statistic#
- value

V$SESSION_EVENT
- sid
- event
- total_waits
- total_timeouts
- time_waited
- average_wait
- max_wait
- time_waited_micro
- event_id

Service specific

V$SERVICE_STATS
- service_name_hash
- service_name
- stat_id
- stat_name
- value

V$SERVICE_EVENT
- service_name
- service_name_hash
- event
- event_id
- total_waits
- total_timeouts
- time_waited
- average_wait
- time_waited_micro

🟧 Cumulative stats
🟦 Wait events

Dynamic Performance Statistics

Statistics must be available for the effective diagnosis of performance problems. The Oracle server generates many types of statistics for different levels of granularity.

At the systemwide level, session level, and service level, both wait events and accumulated statistics are computed. In the slide, the top row of views shows the cumulative statistics. The bottom row shows the wait event views.

When analyzing a performance problem in any of these scopes, you typically look at the change in statistics (delta value) over the period of time you are interested in. All the possible wait events are cataloged in the `V$EVENT_NAME` view. All statistics are cataloged in the `V$STATNAME` view; approximately 480 statistics are available in Oracle Database.

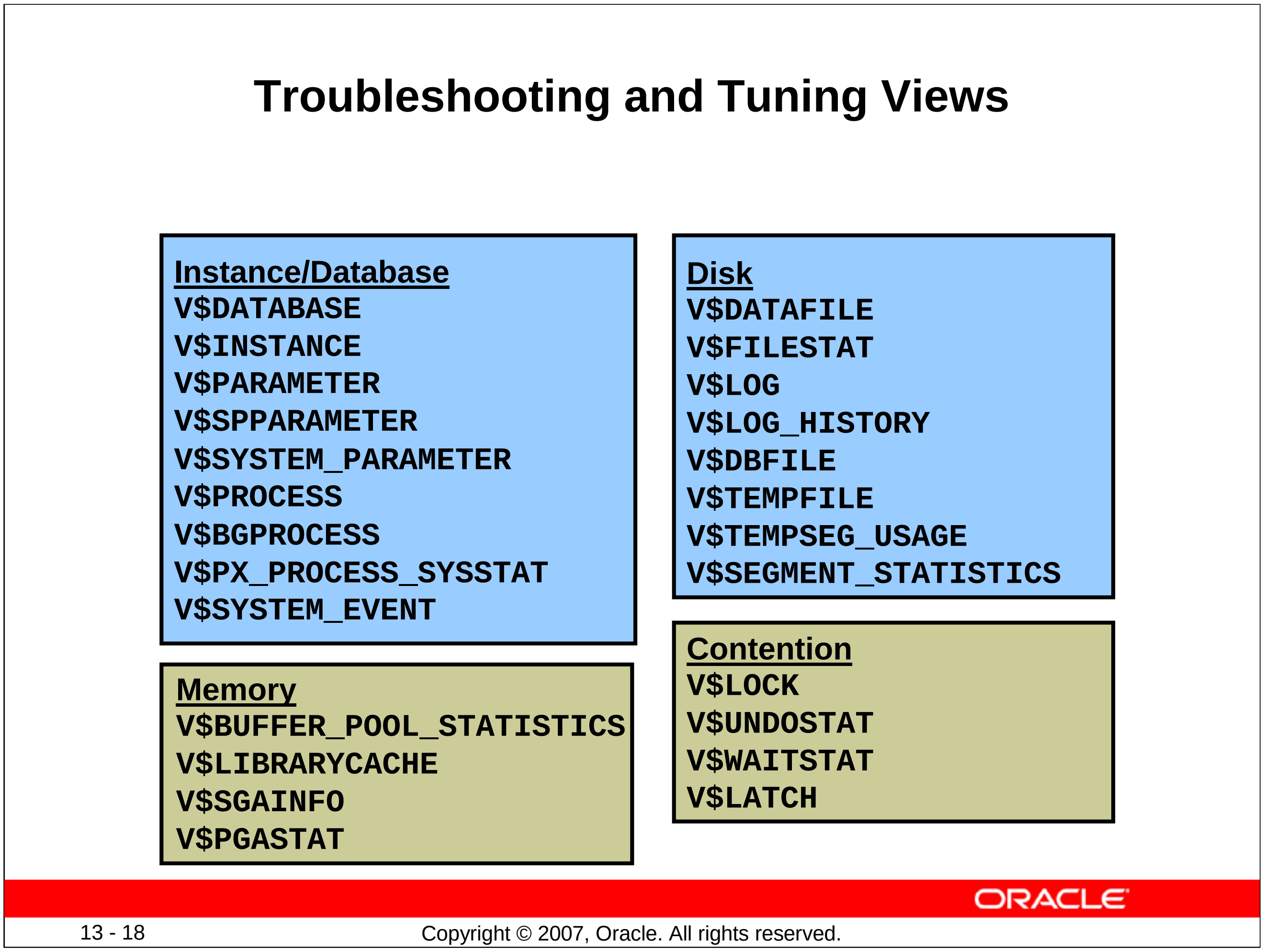
Dynamic Performance Statistics (continued)

Displaying Systemwide Statistics

Example:

```
SQL>  SELECT name, class, value FROM v$sysstat;
NAME                               CLASS      VALUE
-------------------------------  ------ ----------
...
table scans (short tables)           64     135116
table scans (long tables)            64        250
table scans (rowid ranges)           64          0
table scans (cache partitions)       64          3
table scans (direct read)            64          0
table scan rows gotten               64   14789836
table scan blocks gotten             64     558542
...
```

Systemwide statistics are classified by the tuning topic and the debugging purpose. The classes include general instance activity, redo log buffer activity, locking, database buffer cache activity, and so on.



Troubleshooting and Tuning Views

Instance/Database
- V$DATABASE
- V$INSTANCE
- V$PARAMETER
- V$SPPARAMETER
- V$SYSTEM_PARAMETER
- V$PROCESS
- V$BGPROCESS
- V$PX_PROCESS_SYSSTAT
- V$SYSTEM_EVENT

Disk
- V$DATAFILE
- V$FILESTAT
- V$LOG
- V$LOG_HISTORY
- V$DBFILE
- V$TEMPFILE
- V$TEMPSEG_USAGE
- V$SEGMENT_STATISTICS

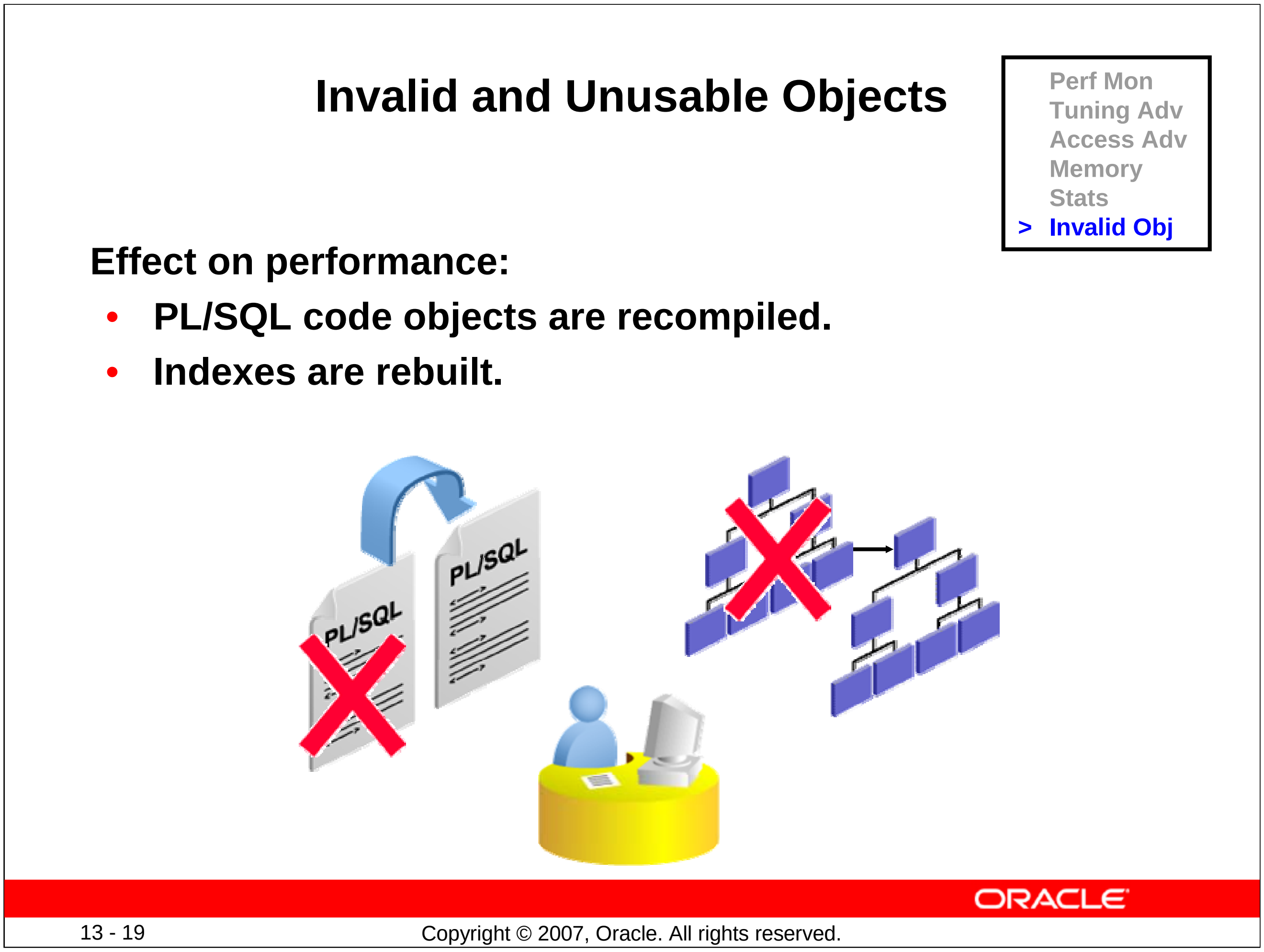
Memory
- V$BUFFER_POOL_STATISTICS
- V$LIBRARYCACHE
- V$SGAINFO
- V$PGASTAT

Contention
- V$LOCK
- V$UNDOSTAT
- V$WAITSTAT
- V$LATCH

Troubleshooting and Tuning Views

The slide lists some of the views that can help you determine the cause of performance problems or analyze the current status of your database.

For a complete description of these views, see the *Oracle Database Reference*.



Invalid and Unusable Objects

Perf Mon
Tuning Adv
Access Adv
Memory
Stats
> Invalid Obj

Effect on performance:

- **PL/SQL code objects are recompiled.**
- **Indexes are rebuilt.**

Invalid and Unusable Objects

Invalid PL/SQL objects and unusable indexes have an impact on performance. Invalid PL/SQL object must be recompiled before they can be used. This requires the compile time to added to the first action that attempts to access the PL/SQL package, procedure, or function. If the PL/SQL does not recompile successfully, the operation fails with an error. Unusable indexes are ignored by the optimizer. If proper performance of a SQL statement depends on a index that has been marked unusable, the performance does not improve until the index is rebuilt.

Invalid PL/SQL objects: The current status of PL/SQL objects can be viewed by querying the data dictionary. You can find invalid PL/SQL objects with:

```
SELECT object_name, object_type FROM DBA_OBJECTS
WHERE status = 'INVALID';
```

By default, the Owner’s Invalid Object Count metric is checked every 24 hours. If the number of objects for an individual owner exceeds two, an alert is issued.

If you find PL/SQL objects with a status of `INVALID`, the first question that you need to answer is “Has this object *ever* been `VALID`?” An application developer often neglects to clean up code that does not work. If the PL/SQL object is invalid because of a code error, there is little that can be done until that error is fixed. If the procedure was valid at some time in the past and has recently become invalid, you have two options for fixing the problem:

Invalid and Unusable Objects (continued)

- Do nothing. Most PL/SQL objects automatically recompile if needed when they are called. Users experience a slight delay while the objects recompile. (In most cases, this delay is not even noticeable.)
- Manually recompile the invalid object.

Invalid PL/SQL objects can be manually recompiled by using Enterprise Manager or through SQL commands:

```
        ALTER PROCEDURE HR.add_job_history COMPILE;
```

Manually recompiling PL/SQL packages requires two steps:

```
        ALTER PACKAGE HR.maintainemp COMPILE;
        ALTER PACKAGE HR.maintainemp COMPILE BODY;
```

Unusable indexes: Invalid indexes can be found by querying the DBA_INDEXES data dictionary view:

```
        SELECT index_name, table_name FROM DBA_INDEXES
        WHERE status = 'UNUSABLE';
```

For partitioned indexes, the status is held in the DBA_IND_PARTITIONS view.

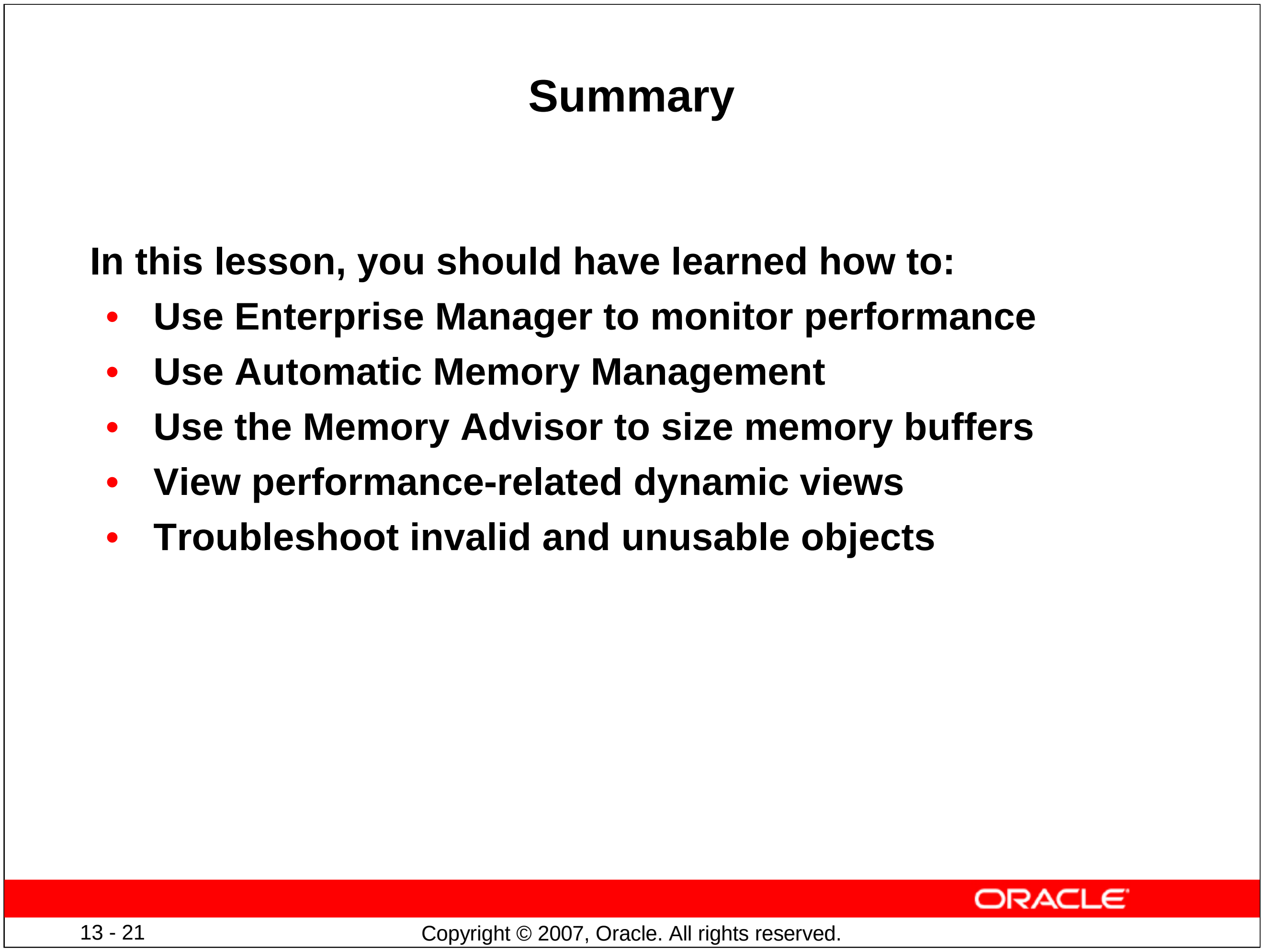
Unusable indexes are made valid by rebuilding them to recalculate the pointers. Rebuilding an unusable index re-creates the index in a new location and then drops the unusable index. This can be done either by using Enterprise Manager or through SQL commands:

```
        ALTER INDEX HR.emp_empid_pk REBUILD;
        ALTER INDEX HR.emp_empid_pk REBUILD ONLINE;
        ALTER INDEX HR.email REBUILD TABLESPACE USERS;
```

If the TABLESPACE clause is left out, the index is rebuilt in the same tablespace where it already exists. The REBUILD ONLINE clause enables users to continue updating the index's table while the rebuild takes place. (Without the ONLINE keyword, users must wait for the rebuild to finish before performing DML on the affected table. If the index is unusable, it is not used during the rebuild even if the ONLINE keyword is used.)

Enterprise Manager uses the Reorganize action to repair an UNUSABLE index.

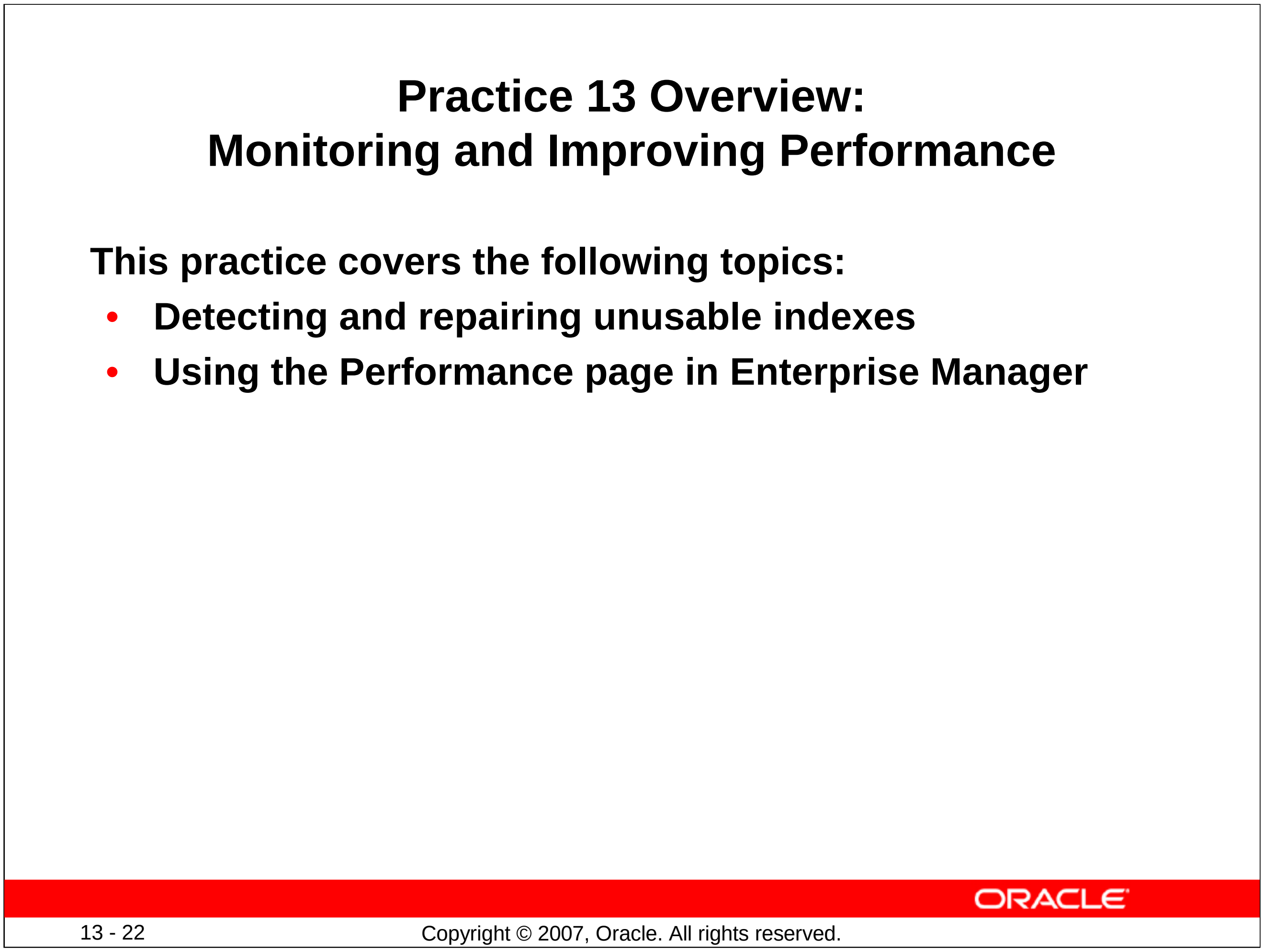
Note: Rebuilding an index requires that free space be available for the rebuild. Verify that there is sufficient space before attempting the rebuild. Enterprise Manager automatically checks space requirements.



Summary

In this lesson, you should have learned how to:

- Use Enterprise Manager to monitor performance
- Use Automatic Memory Management
- Use the Memory Advisor to size memory buffers
- View performance-related dynamic views
- Troubleshoot invalid and unusable objects



Practice 13 Overview: Monitoring and Improving Performance

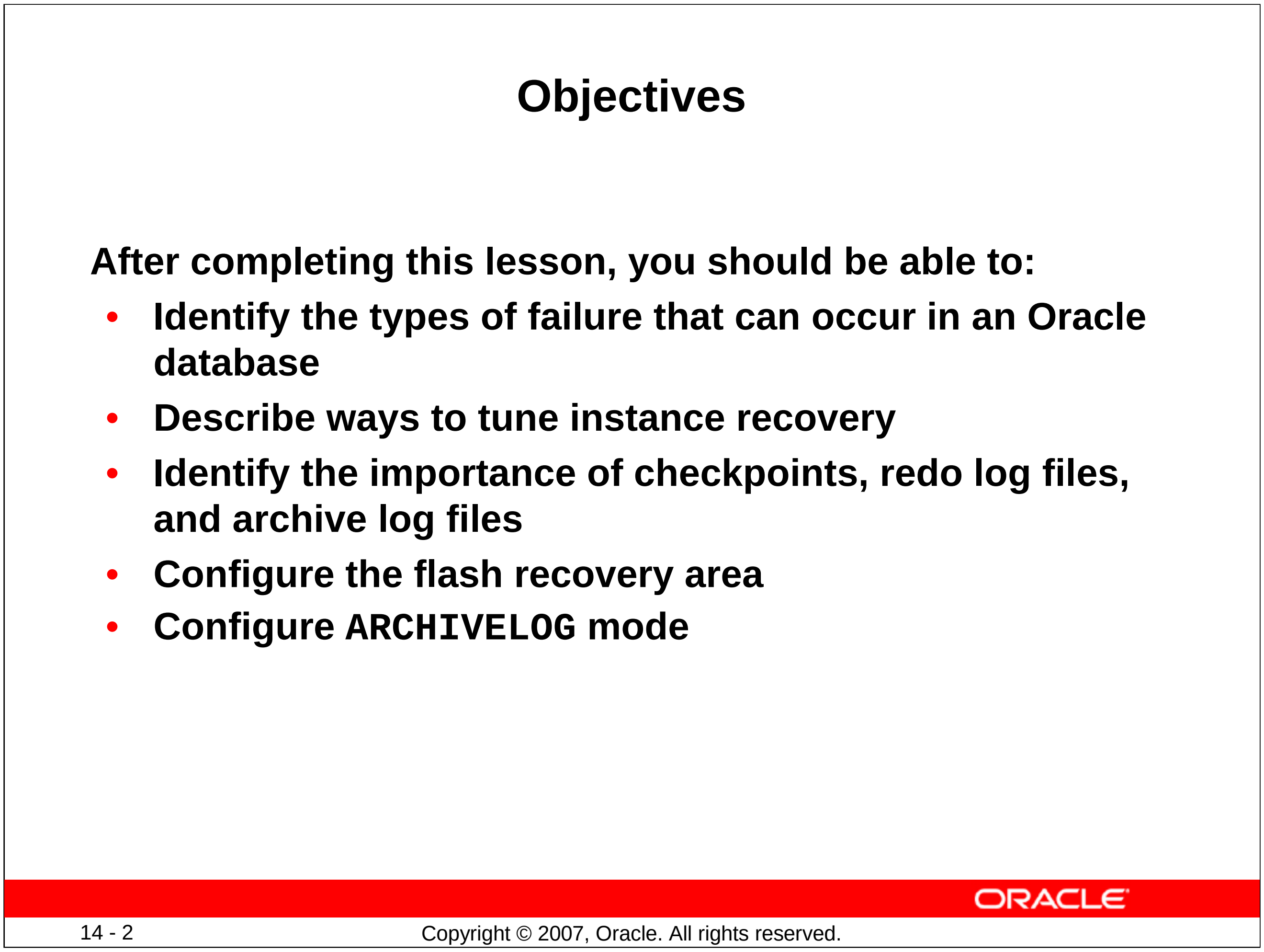
This practice covers the following topics:

- Detecting and repairing unusable indexes
- Using the Performance page in Enterprise Manager

13 - 22

14

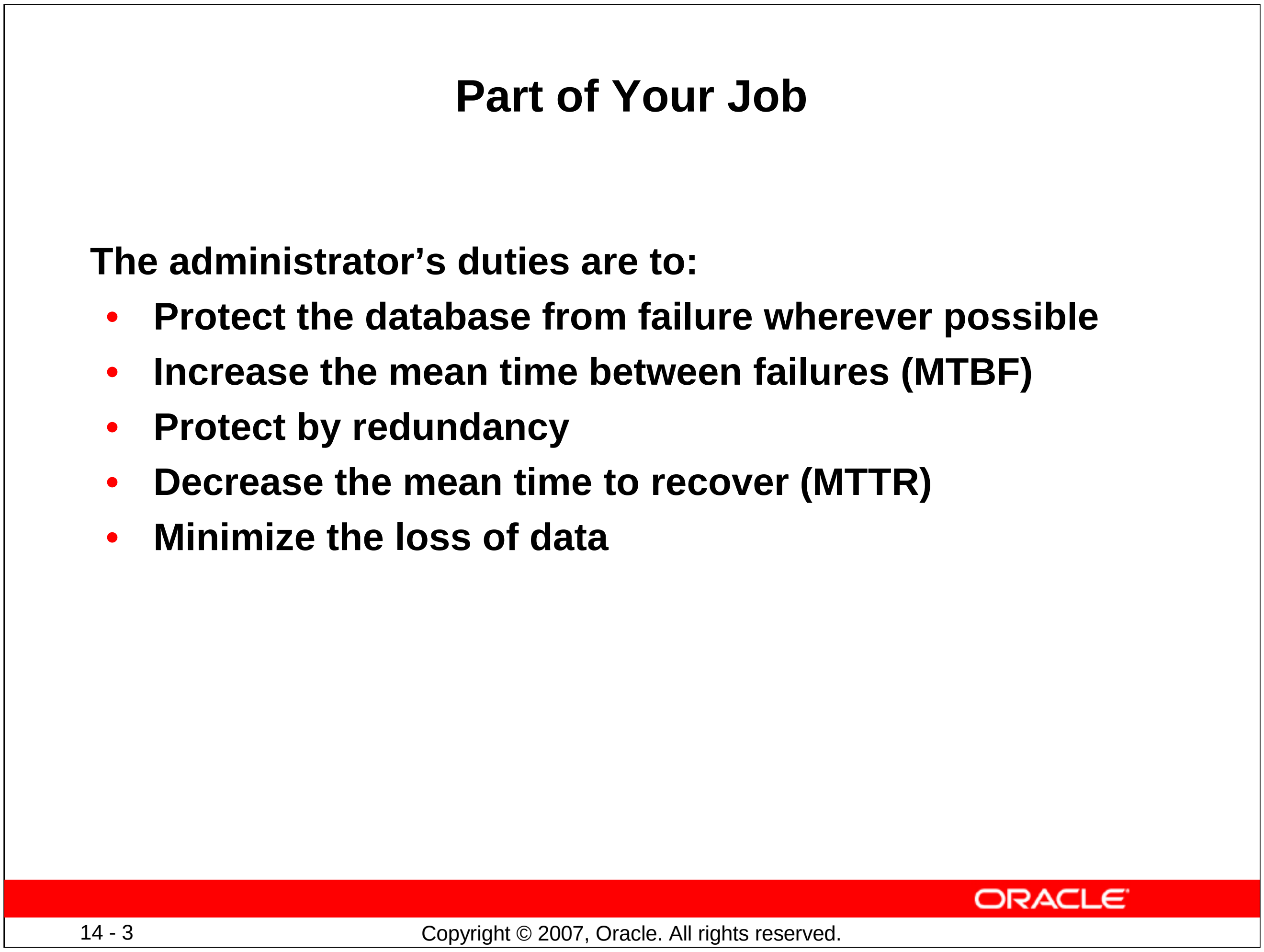
Backup and Recovery Concepts



Objectives

After completing this lesson, you should be able to:

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



Part of Your Job

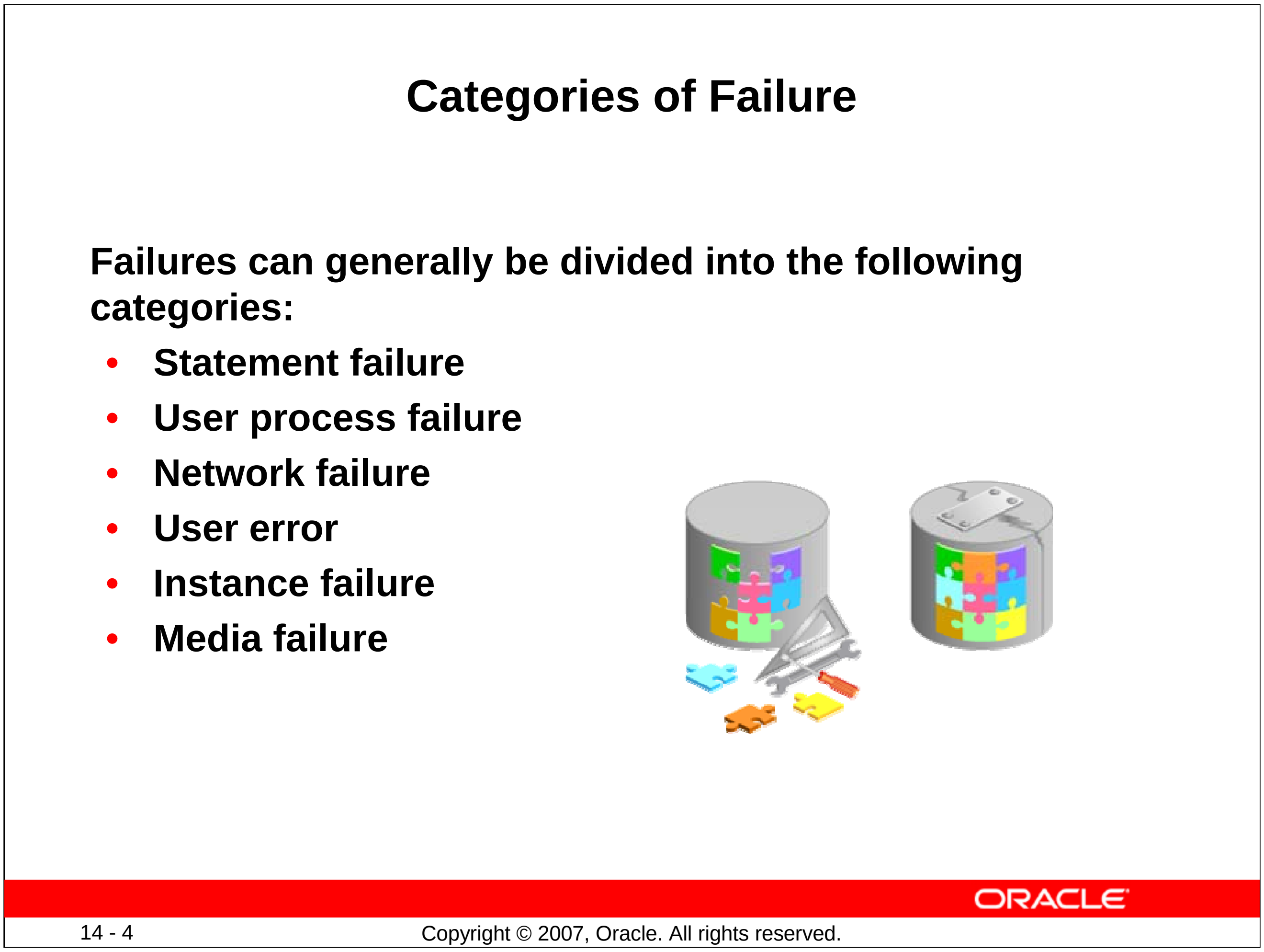
The administrator's duties are to:

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

Part of Your Job

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

- Anticipates and works to avoid common causes of failure
- Works to increase the mean time between failures (MTBF) that negatively affect availability
- Ensures that hardware is as reliable as possible, that critical components are protected by redundancy, and that operating system maintenance is performed in a timely manner. The Oracle database provides advanced configuration options to increase MTBF, including:
  - Real Application Clusters (discussed in the *Oracle Database 11g: Real Application Clusters* course)
  - Streams (discussed in the *Oracle Database 11g: Implement Streams* course)
- Decreases the mean time to recover (MTTR) by practicing recovery procedures in advance and configuring backups so that they are readily available when needed
- Minimizes the loss of data. DBAs who follow accepted best practices can configure their databases so that no committed transaction is ever lost. Entities that assist in guaranteeing this include:
  - Archive log files (discussed later in this lesson)
  - Standby databases and Oracle Data Guard (discussed in the *Oracle Database 11g: Data Guard Administration* course)



Categories of Failure

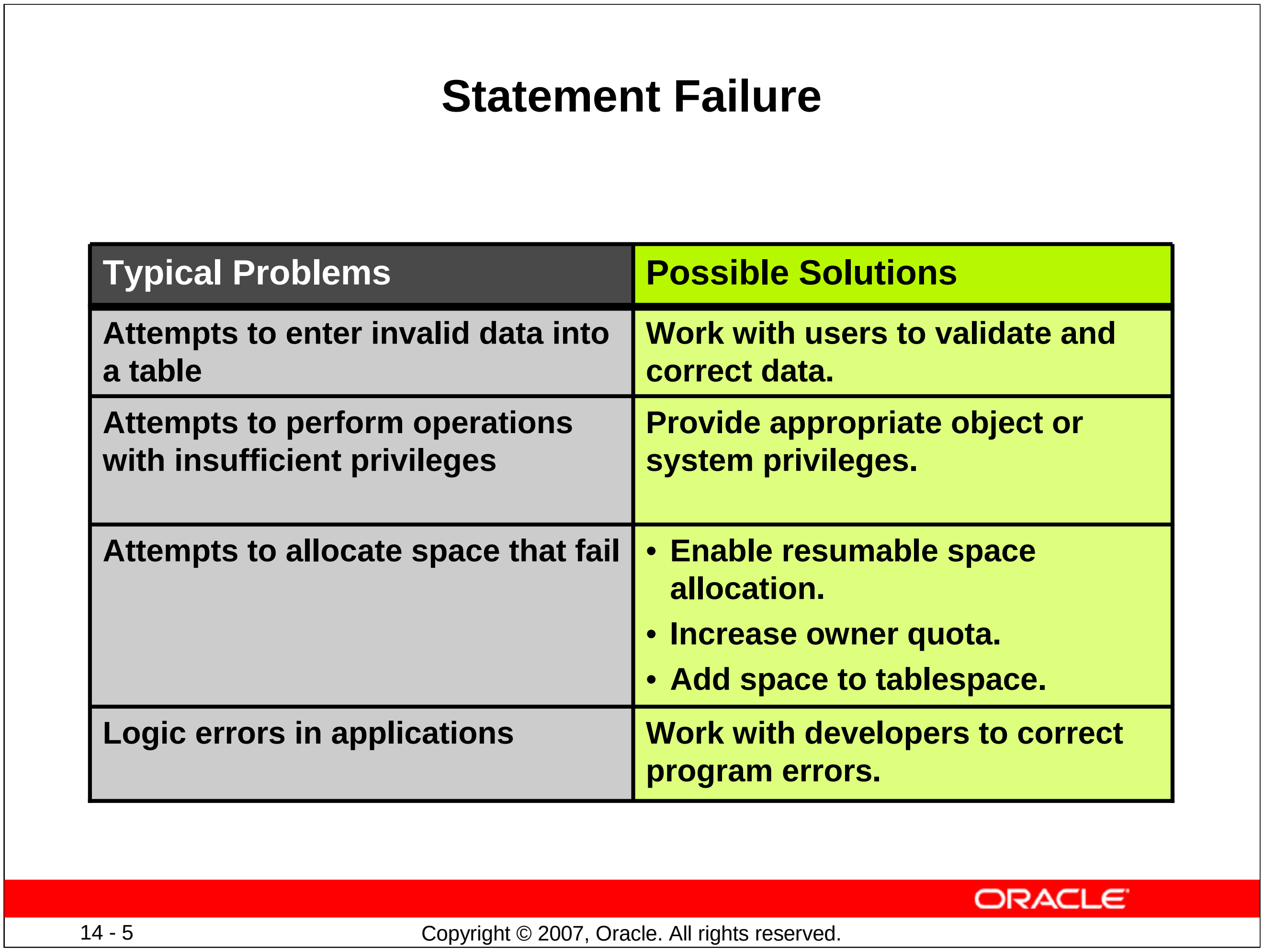
Failures can generally be divided into the following categories:

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

14 - 4

Categories of Failure

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



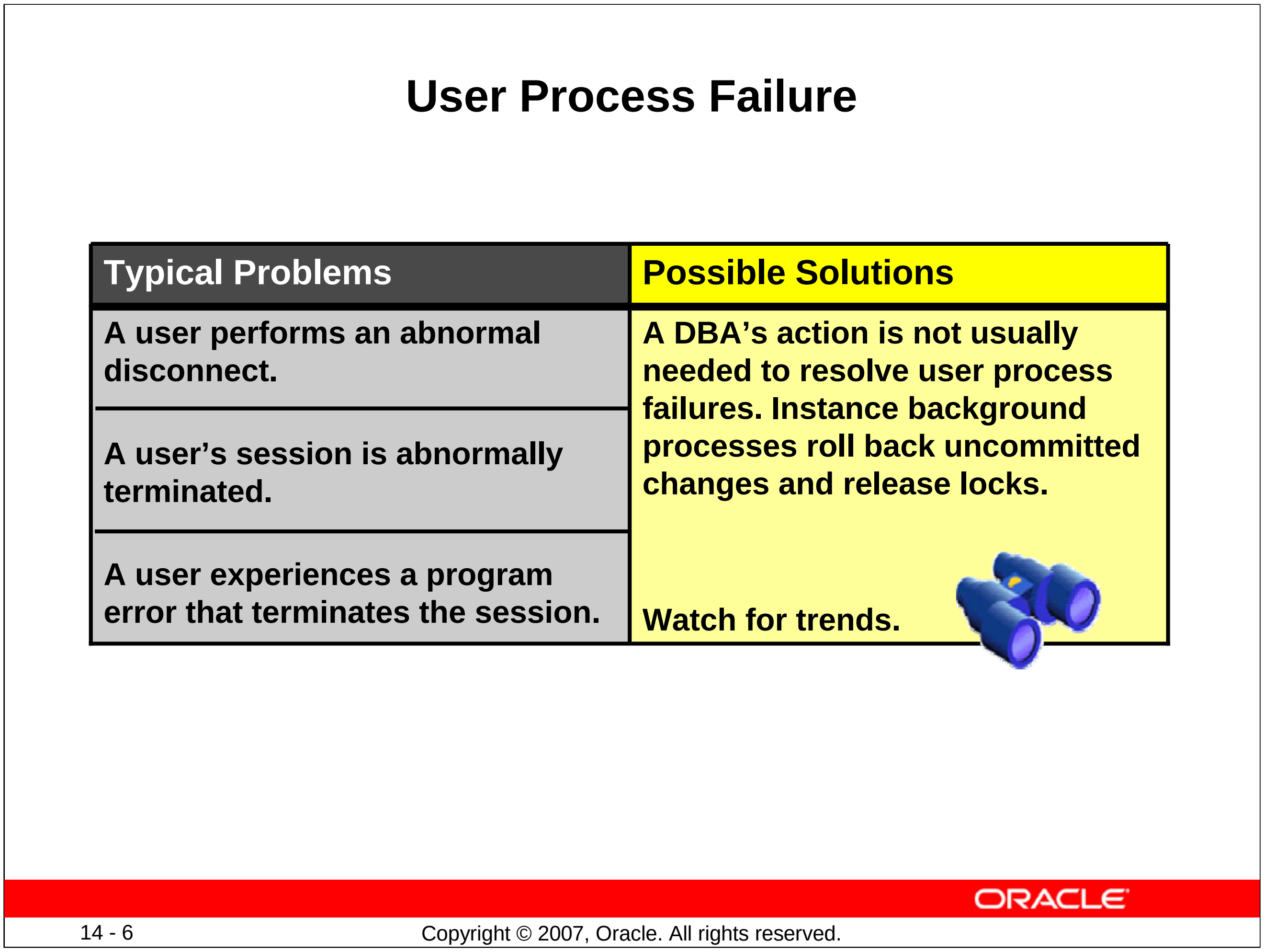
Statement Failure

Typical Problems	Possible Solutions
Attempts to enter invalid data into a table	Work with users to validate and correct data.
Attempts to perform operations with insufficient privileges	Provide appropriate object or system privileges.
Attempts to allocate space that fail	• Enable resumable space allocation. • Increase owner quota. • Add space to tablespace.
Logic errors in applications	Work with developers to correct program errors.

Statement Failure

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

To understand logic errors in applications, you can use LogMiner (select Enterprise Manager > Availability > View and Manage Transactions, or access from the command line) to examine the redo of transactions.



User Process Failure

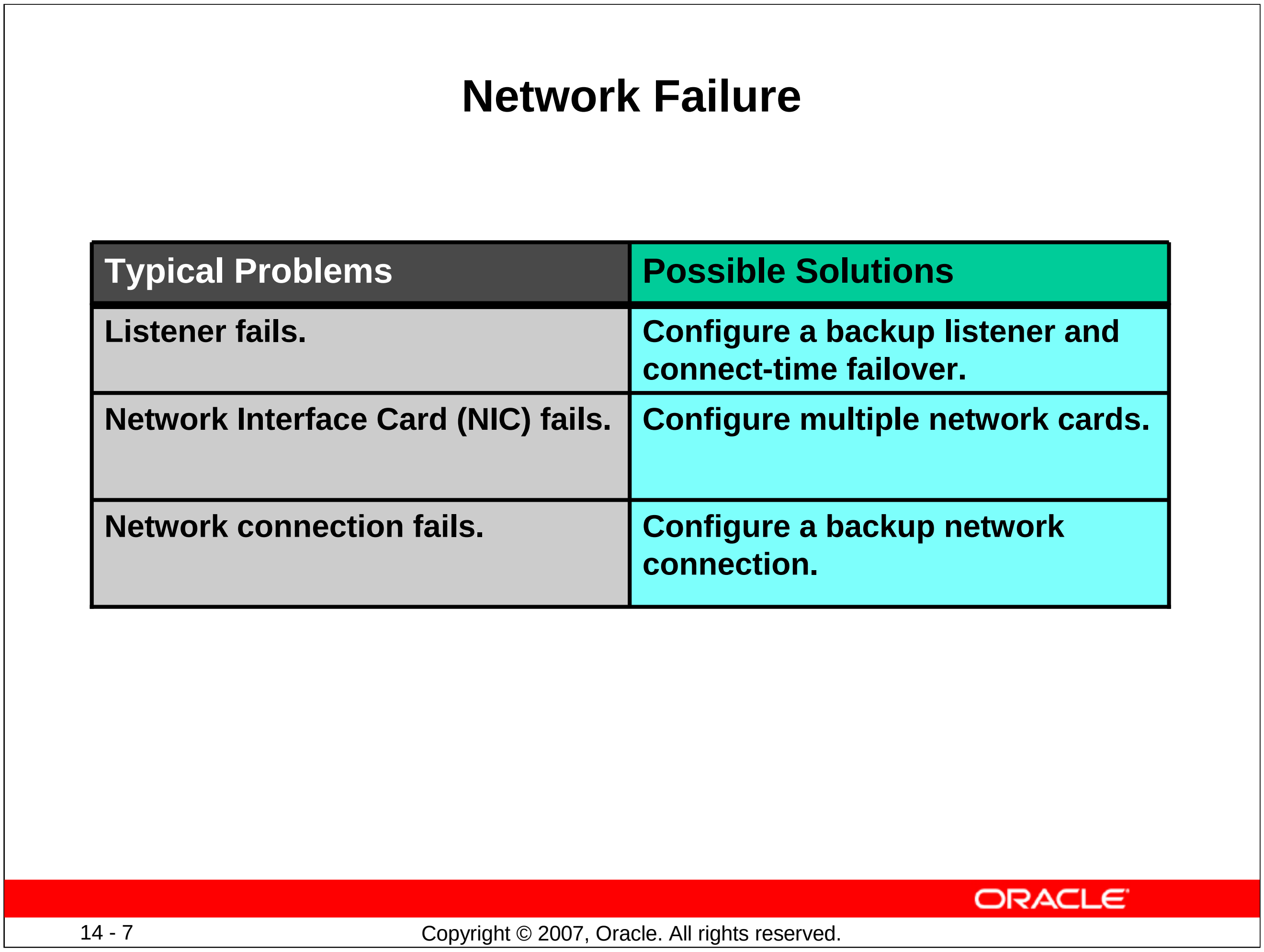
Typical Problems	Possible Solutions
A user performs an abnormal disconnect.	A DBA’s action is not usually needed to resolve user process failures. Instance background processes roll back uncommitted changes and release locks. Watch for trends.
A user’s session is abnormally terminated.	
A user experiences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

User Process Failure

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

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

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



Network Failure

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

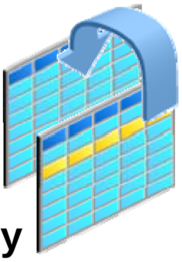
Network Failure

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

User Error

Using Flashback technology:

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



For error analysis:

- Oracle Flashback Query (`SELECT … AS OF…`)
- Oracle Flashback Versions Query (`SELECT … VERSIONS BETWEEN…`)
- Oracle Flashback Transaction Query

For error recovery:

- Oracle Flashback Transaction Backout
- Oracle Flashback Table
- Oracle Flashback Drop

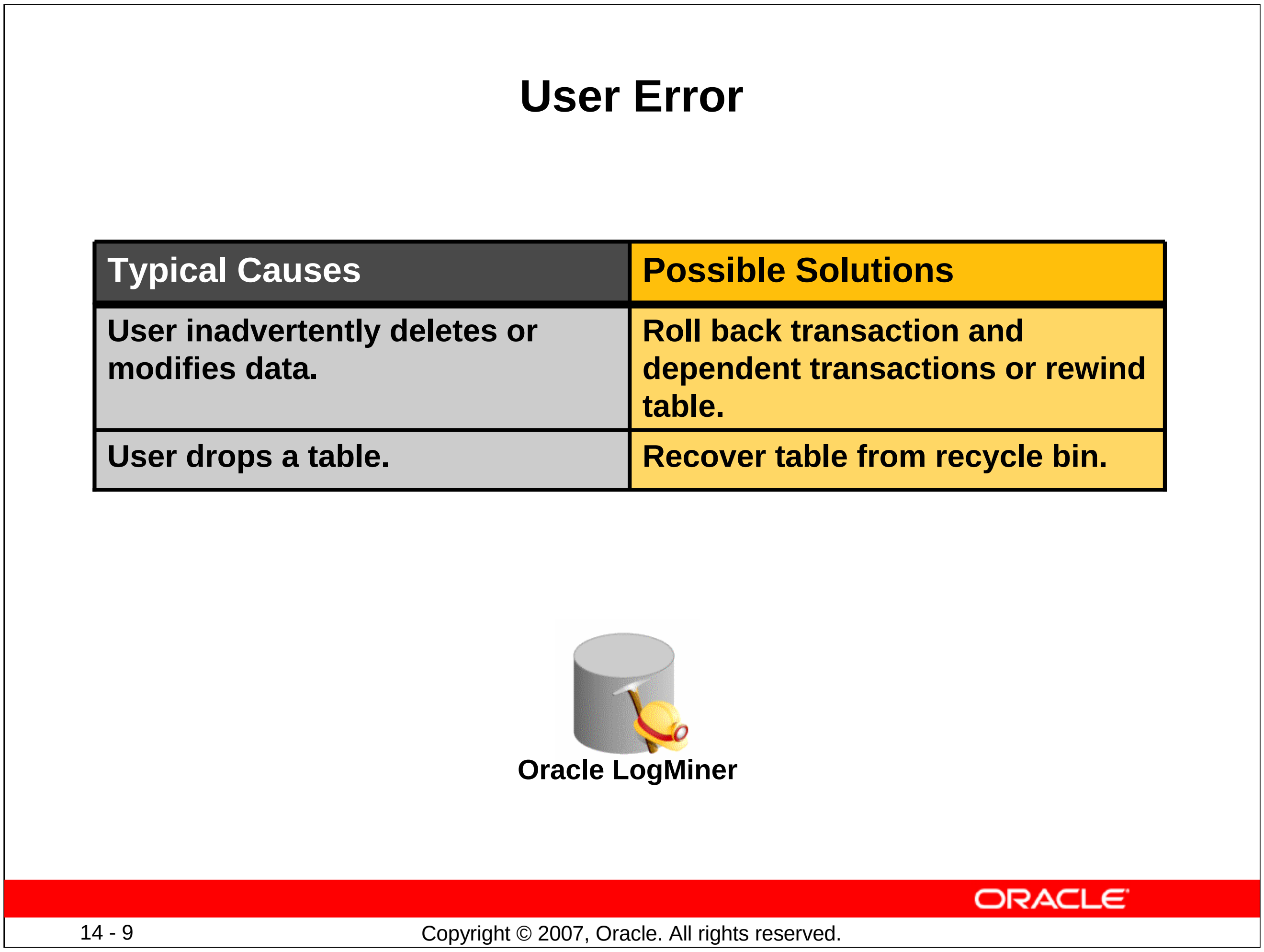
User Error

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

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

Possible solutions to recover from user error:

- **Flashback Transaction Backout:** Rolls back a specific transaction and dependent transactions
- **Flashback Table:** Rewinds one or more tables to their contents at a previous time without affecting other database objects
- **Flashback Drop:** Reverses the effects of dropping a table by returning the dropped table from the recycle bin to the database along with dependent objects such as indexes and triggers



User Error

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

Oracle LogMiner
ORACLE

14 - 9

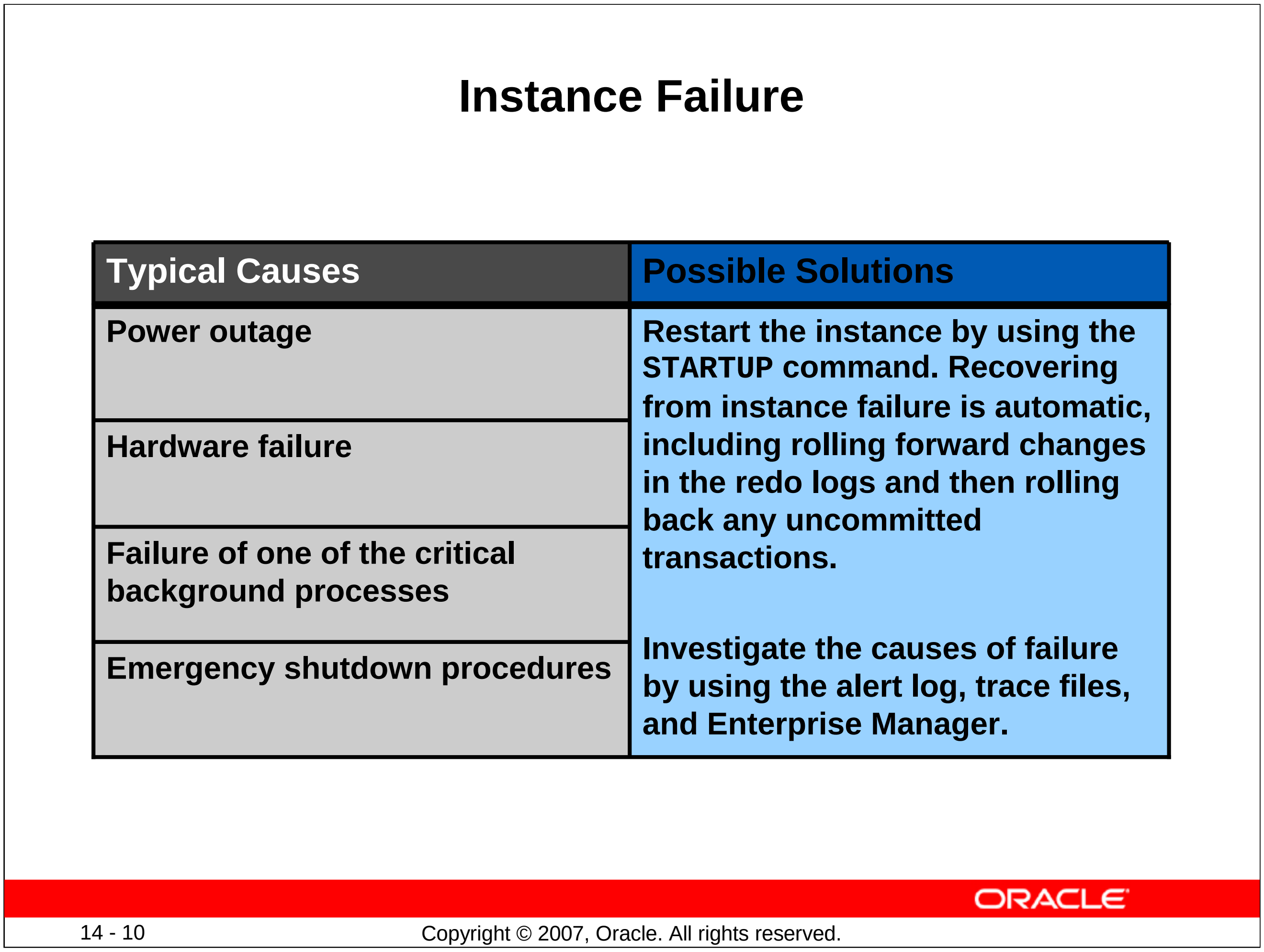
User Error (continued)

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

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

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

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



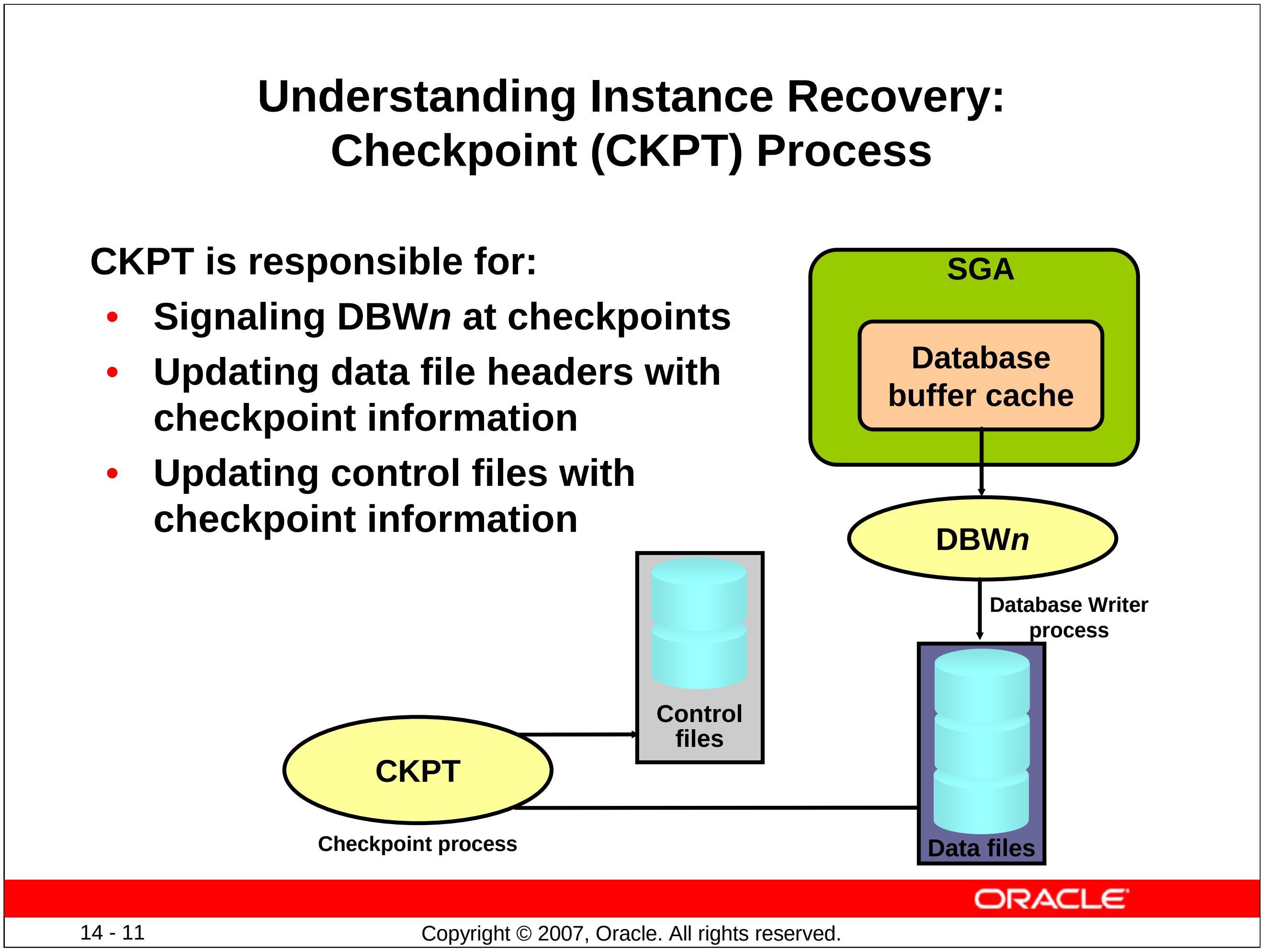
Instance Failure

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

Instance Failure

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

Administrator involvement in recovering from instance failure is usually limited to restarting the instance and working to prevent future occurrences.



Understanding Instance Recovery: Checkpoint (CKPT) Process

CKPT is responsible for:

- **Signaling DBW*n* at checkpoints**
- **Updating data file headers with checkpoint information**
- **Updating control files with checkpoint information**

Understanding Instance Recovery: Checkpoint (CKPT) Process

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

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

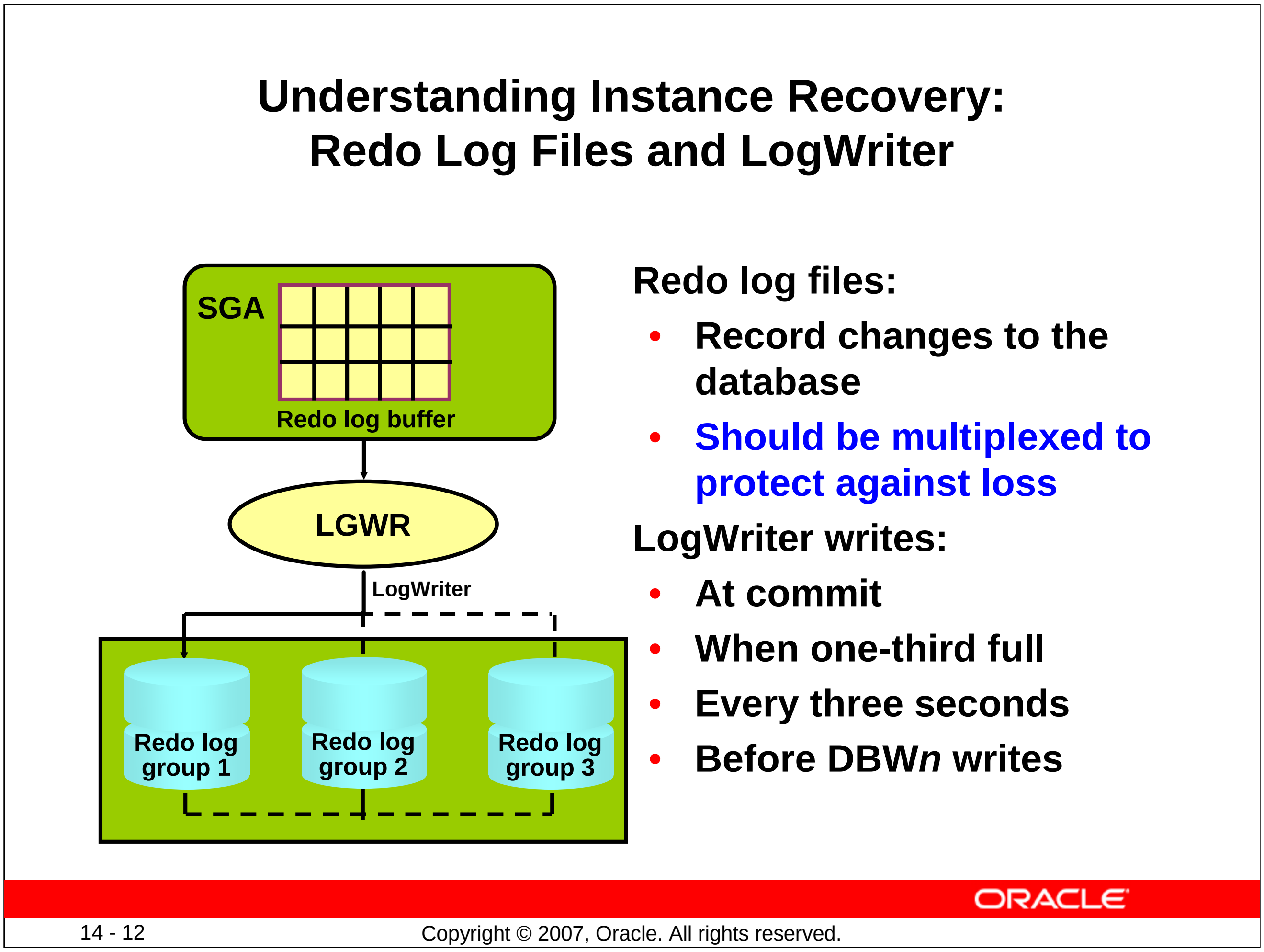
In the event of a log switch, the CKPT process also writes this checkpoint information to the headers of data files.

Checkpoints exist for the following reasons:

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

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

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



Understanding Instance Recovery: Redo Log Files and LogWriter

Redo log files:

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

LogWriter writes:

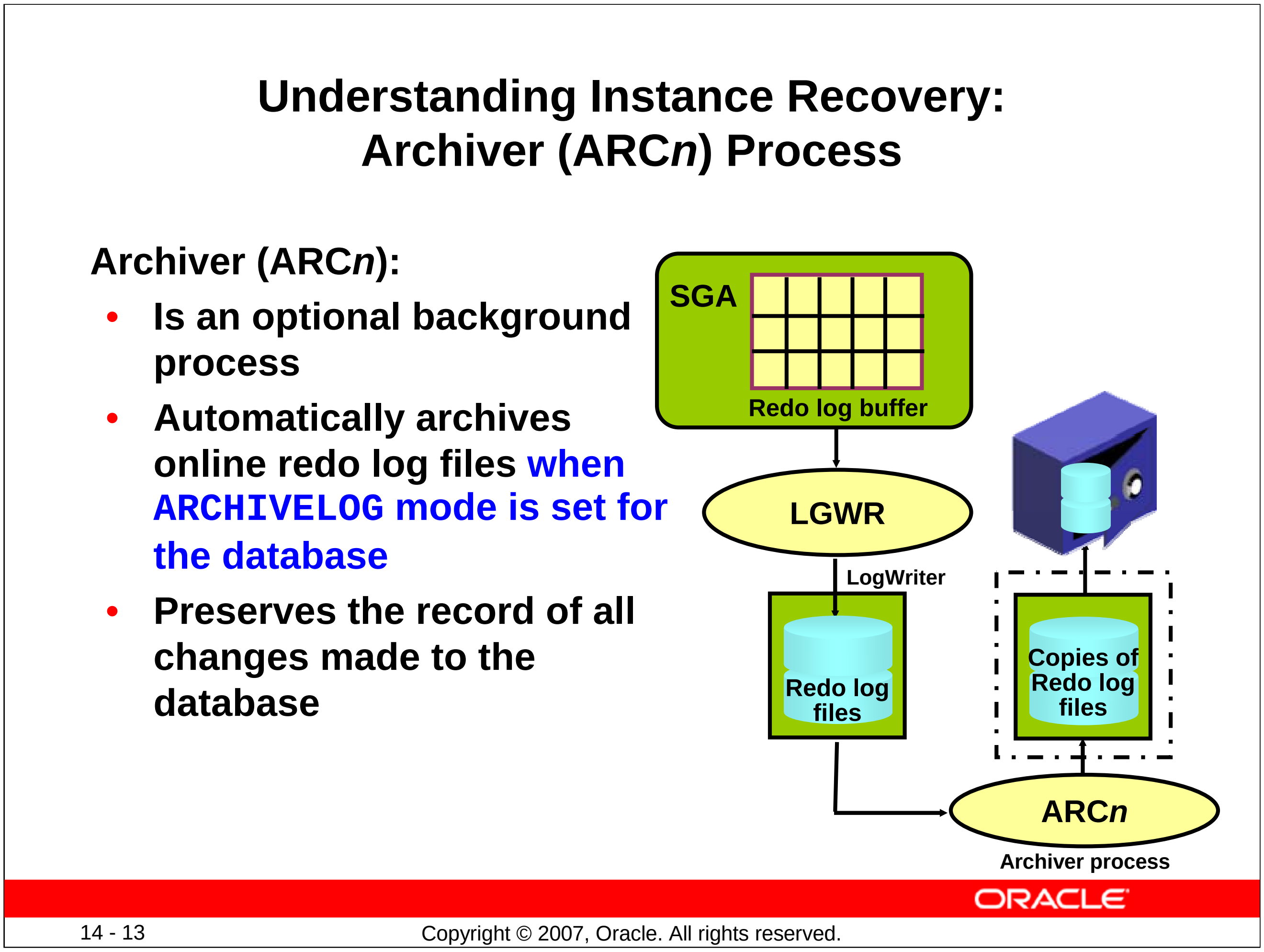
- At commit
- When one-third full
- Every three seconds
- Before DBW*n* writes

Background Processes and Recovery: Redo Log Files and LogWriter

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

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

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



Understanding Instance Recovery: Archiver (ARC*n*) Process

Archiver (ARC*n*):

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

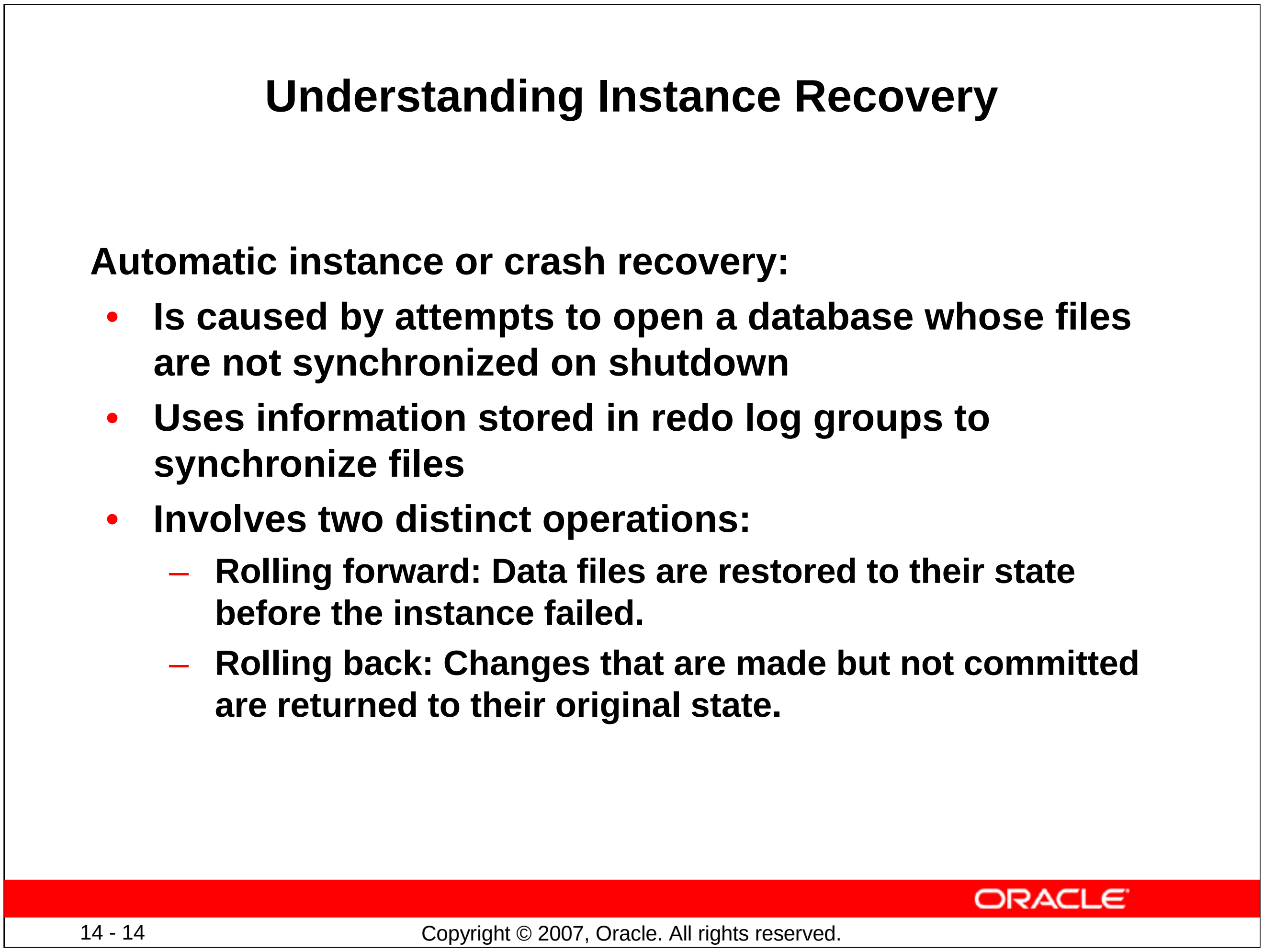
Understanding Instance Recovery: The Archiver (ARC*n*) Process

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

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

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

Note: `ARCHIVELOG` mode is essential for most backup strategies (and is very easy to configure).



Understanding Instance Recovery

Automatic instance or crash recovery:

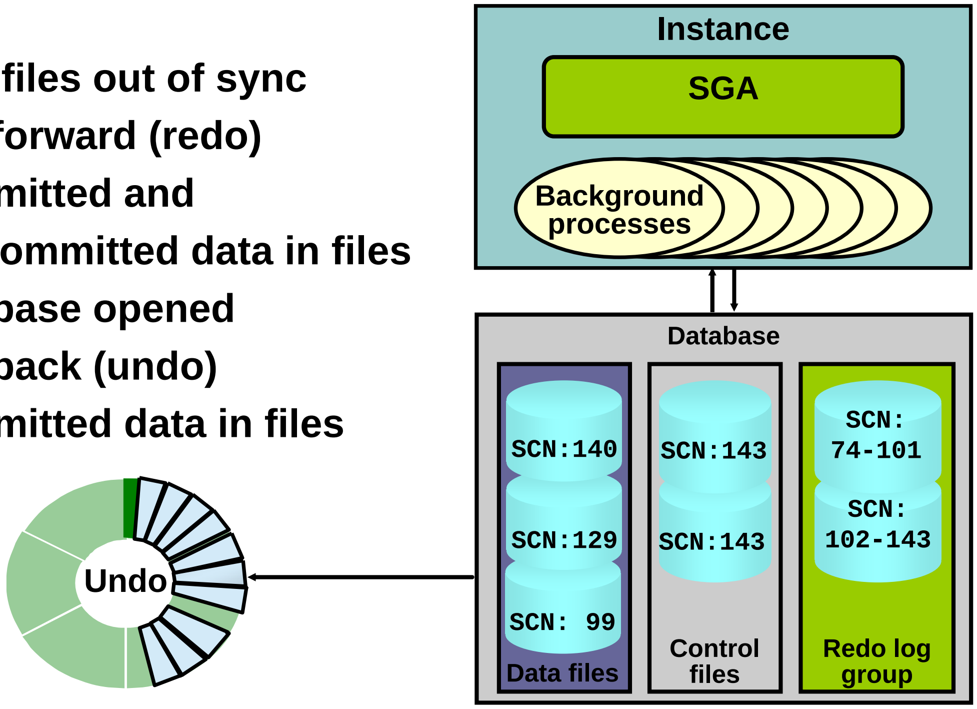
- **Is caused by attempts to open a database whose files are not synchronized on shutdown**
- **Uses information stored in redo log groups to synchronize files**
- **Involves two distinct operations:**
  - **Rolling forward: Data files are restored to their state before the instance failed.**
  - **Rolling back: Changes that are made but not committed are returned to their original state.**

Instance Recovery

The Oracle database automatically recovers from instance failure. All that the DBA needs to do is start the instance normally. The instance mounts the control files and then attempts to open the data files. When it discovers that the data files have not been synchronized during shutdown, the instance uses information contained in the redo log groups to roll the data files forward to the time of shutdown. Then the database is opened and (because the undo tablespace is also rolled forward) any uncommitted transactions are rolled back.

Phases of Instance Recovery

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

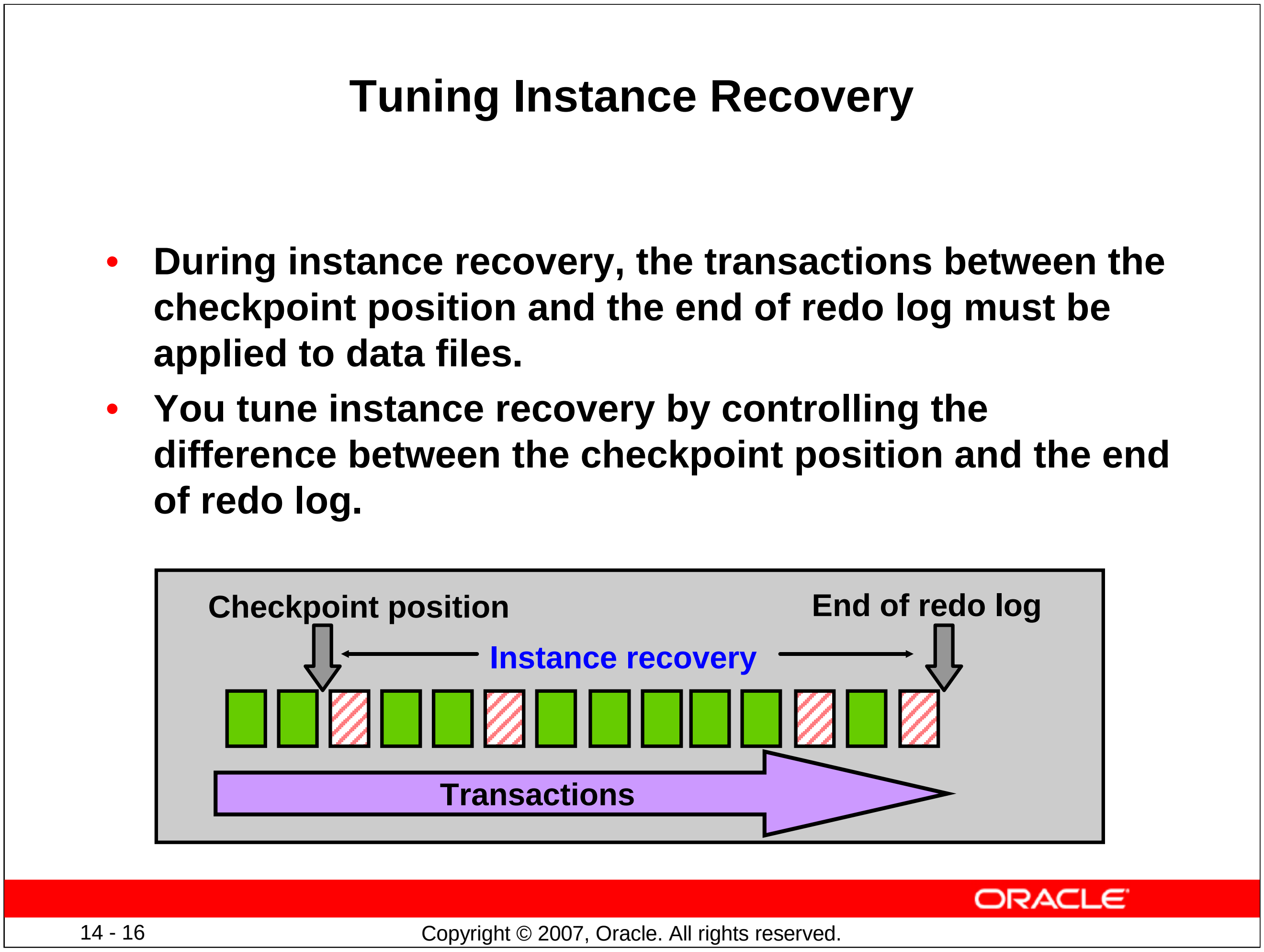


Phases of Instance Recovery

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

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

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



Tuning Instance Recovery

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

Tuning Instance Recovery

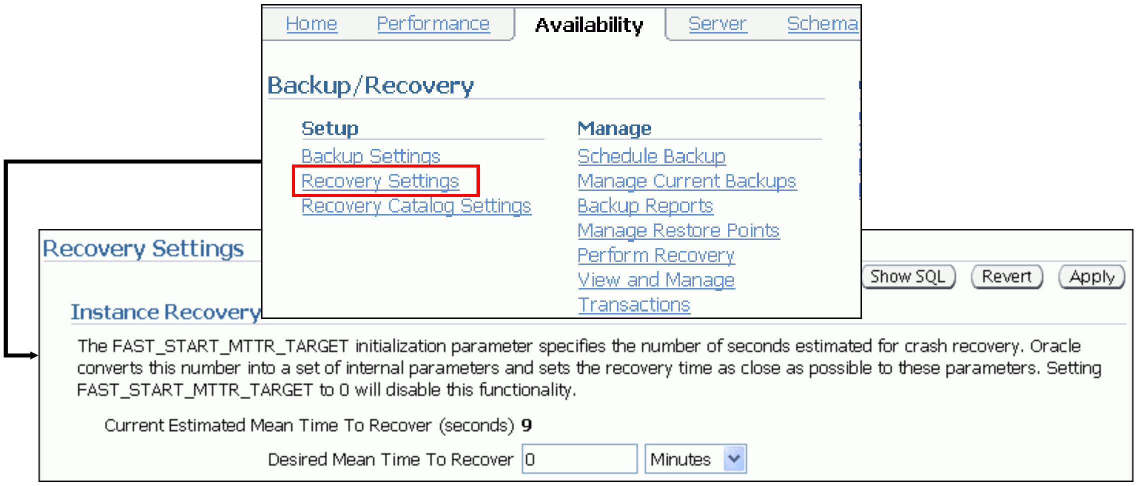
Transaction information is recorded in the redo log groups before the instance returns `commit complete` for a transaction. The information in the redo log groups guarantees that the transaction can be recovered in case of a failure. The transaction information also needs to be written to the data file.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

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

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

Using the MTTR Advisor

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



The FAST_START_MTTR_TARGET initialization parameter specifies the number of seconds estimated for crash recovery. Oracle converts this number into a set of internal parameters and sets the recovery time as close as possible to these parameters. Setting FAST_START_MTTR_TARGET to 0 will disable this functionality.

Current Estimated Mean Time To Recover (seconds) **9**

Desired Mean Time To Recover 0 Minutes

Using the MTTR Advisor

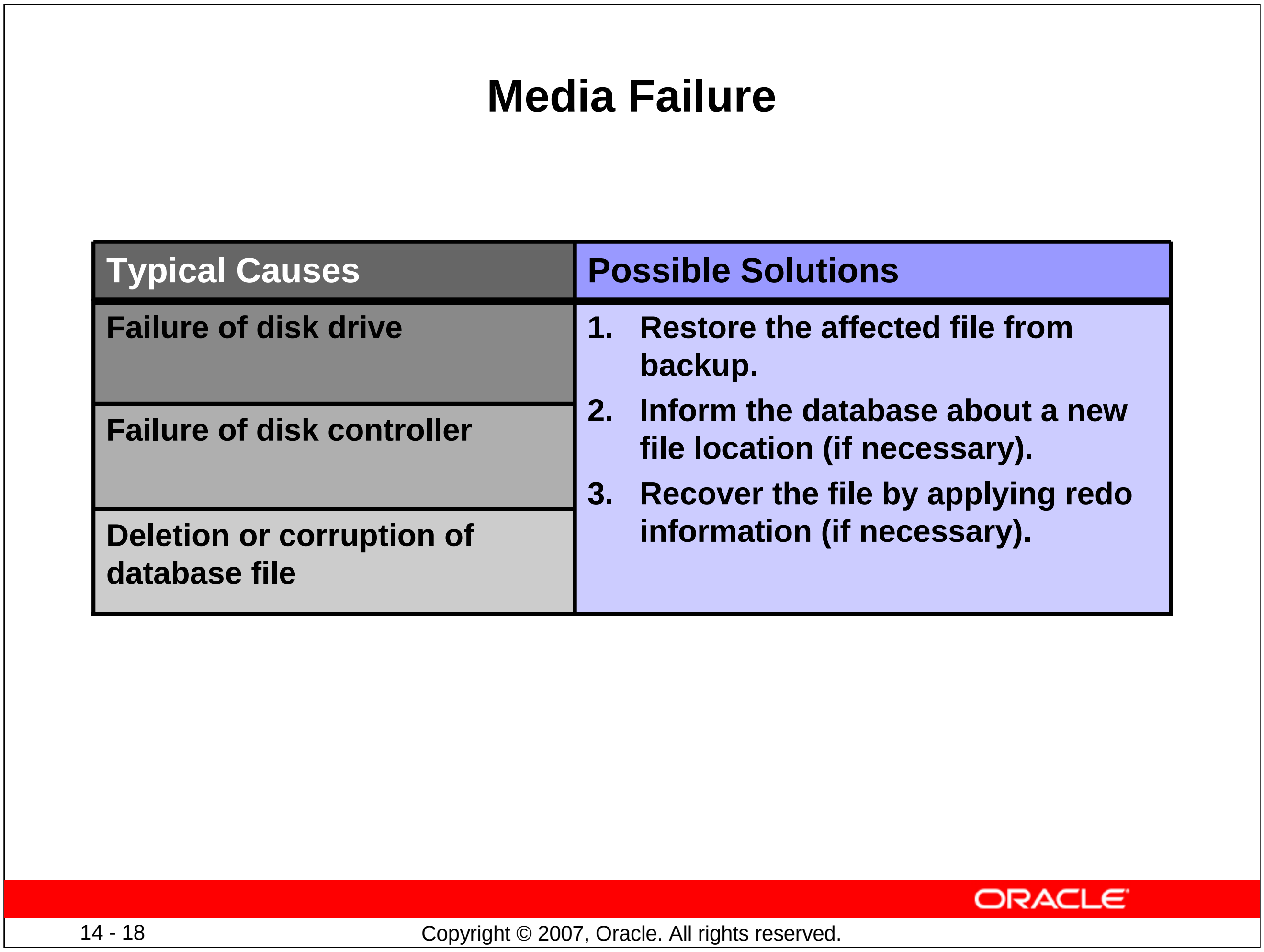
For assistance in setting the MTTR target, select either of the following:

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

The MTTR Advisor converts the FAST_START_MTTR_TARGET value into several parameters to enable instance recovery in the desired time (or as close to it as possible).

Explicit setting of the FAST_START_MTTR_TARGET parameter to 0 disables automatic checkpoint tuning. Explicit setting of the FAST_START_MTTR_TARGET parameter to a value other than 0 also enables the MTTR Advisor. (When specified, FAST_START_MTTR_TARGET is overridden by LOG_CHECKPOINT_INTERVAL.)

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



Media Failure

Typical Causes	Possible Solutions
Failure of disk drive	1. Restore the affected file from backup. 2. Inform the database about a new file location (if necessary). 3. Recover the file by applying redo information (if necessary).
Failure of disk controller	
Deletion or corruption of database file	

Media Failure

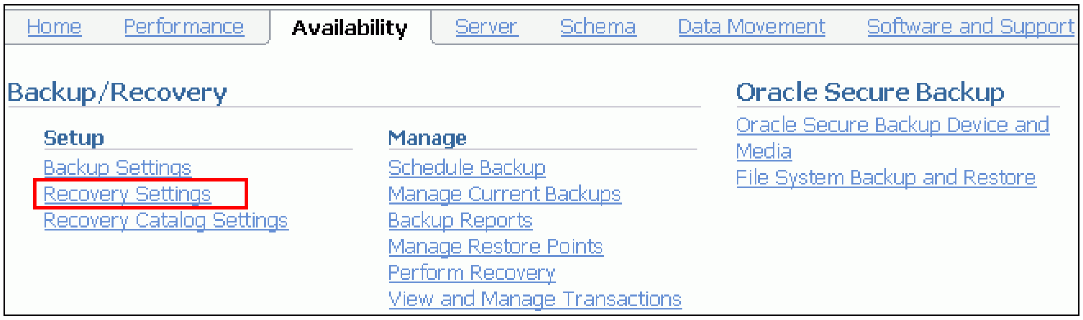
Oracle Corporation defines *media failure* as any failure that results in the loss or corruption of one or more database files (data, control, or redo log file).

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

Configuring for Recoverability

To configure your database for maximum recoverability, you must:

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

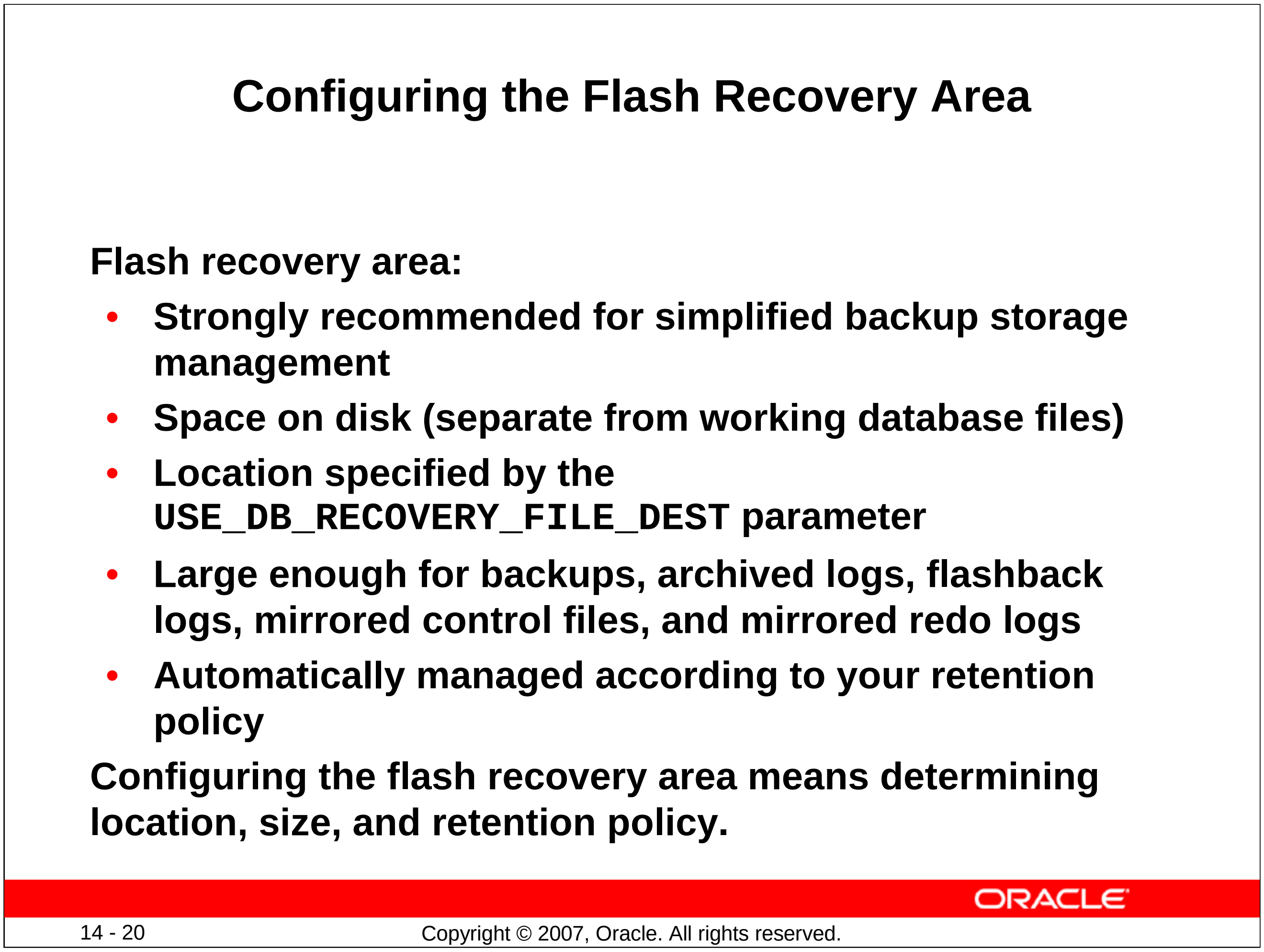


Configuring for Recoverability

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

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

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



Configuring the Flash Recovery Area

Flash recovery area:

- **Strongly recommended for simplified backup storage management**
- **Space on disk (separate from working database files)**
- **Location specified by the `USE_DB_RECOVERY_FILE_DEST` parameter**
- **Large enough for backups, archived logs, flashback logs, mirrored control files, and mirrored redo logs**
- **Automatically managed according to your retention policy**

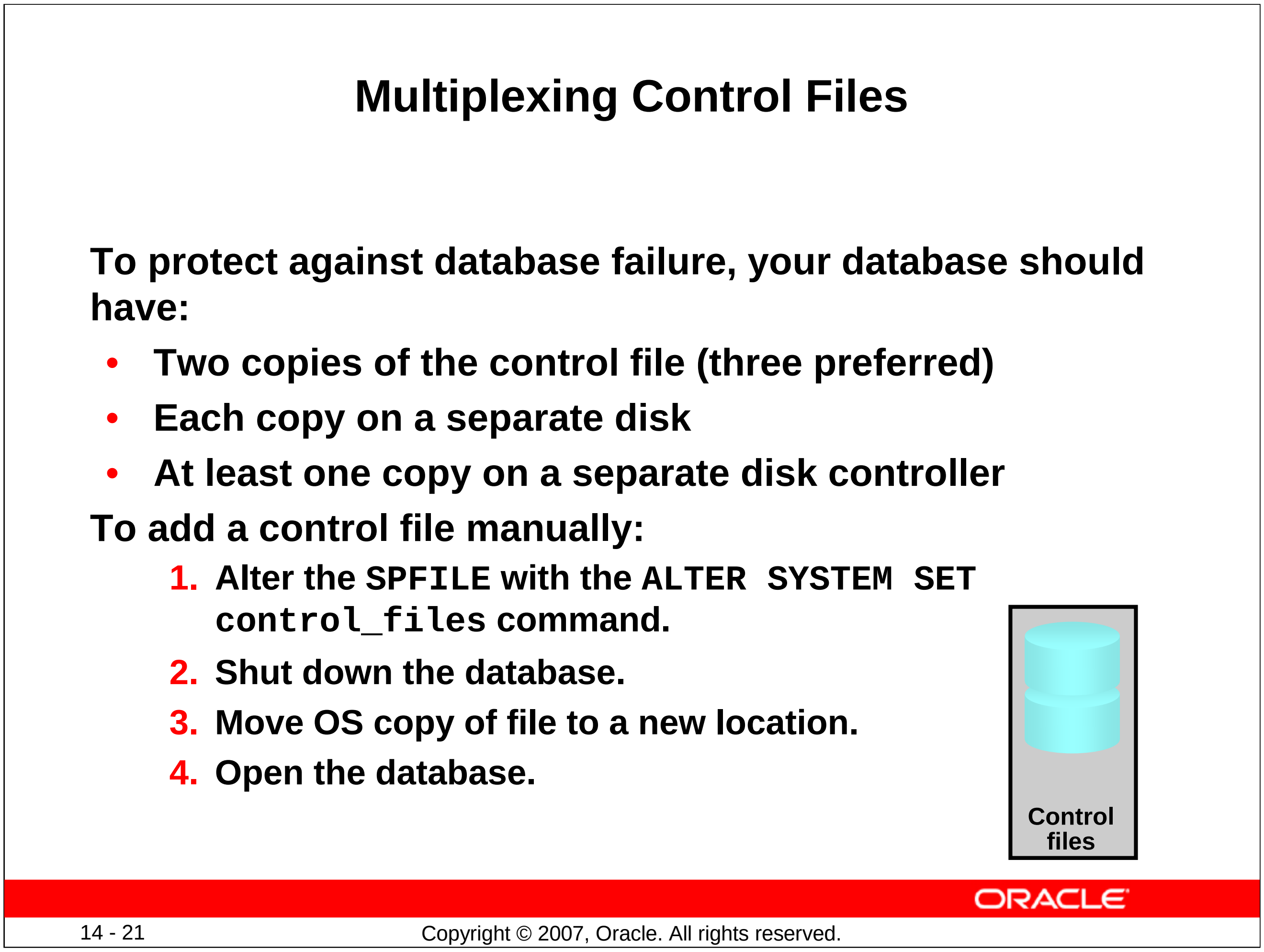
Configuring the flash recovery area means determining location, size, and retention policy.

Configuring the Flash Recovery Area

The flash recovery area is a space that is set aside on the disk to contain archived logs, backups, flashback logs, mirrored control files, and mirrored redo logs. A flash recovery area simplifies backup storage management and is strongly recommended. You should place the flash recovery area on a disk that is separate from the working set of database files. Otherwise, the disk becomes a single point of failure for your database.

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

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



Multiplexing Control Files

To protect against database failure, your database should have:

- **Two copies of the control file (three preferred)**
- **Each copy on a separate disk**
- **At least one copy on a separate disk controller**

To add a control file manually:

1. **Alter the `SPFILE` with the `ALTER SYSTEM SET control_files` command.**
2. **Shut down the database.**
3. **Move OS copy of file to a new location.**
4. **Open the database.**

Multiplexing Control Files

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

If your database is created with the Database Configuration Assistant (DBCA) using Oracle Managed Files (OMF), you have two control files. If you do not use OMF, there are three control files.

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

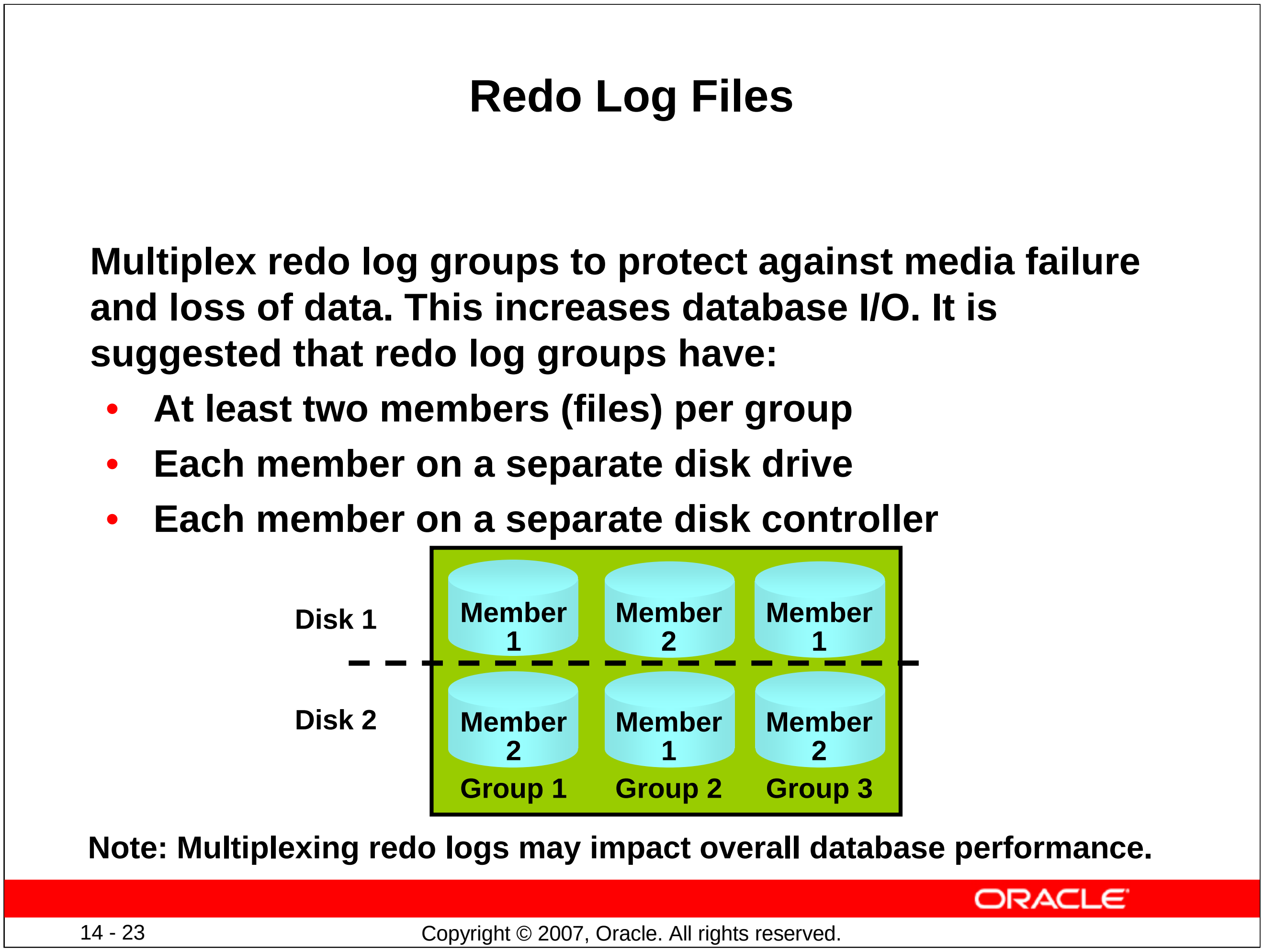
Multiplexing Control Files (continued)

Adding a Control File

In an OMF database, all control files must be re-created (so the following steps do not apply).

In other databases, adding a control file is a manual operation:

1. Alter the `SPFILE` with the following command:
   ```
   ALTER SYSTEM SET control_files =
   '/u01/app/oracle/oradata/orcl/control01.ctl' ,
   '/u01/app/oracle/oradata/orcl/control02.ctl' ,
   '/u01/app/oracle/oradata/orcl/control03.ctl' SCOPE=SPFILE;
   ```
2. Shut down the database.
3. Use the operating system to copy an existing control file to the location you select for your new file.
4. Open the database.



Redo Log Files

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

- **At least two members (files) per group**
- **Each member on a separate disk drive**
- **Each member on a separate disk controller**

Disk 1
Disk 2
Member 1
Member 2
Member 1
Member 2
Member 1
Member 2
Group 1
Group 2
Group 3

Note: Multiplexing redo logs may impact overall database performance.

14 - 23

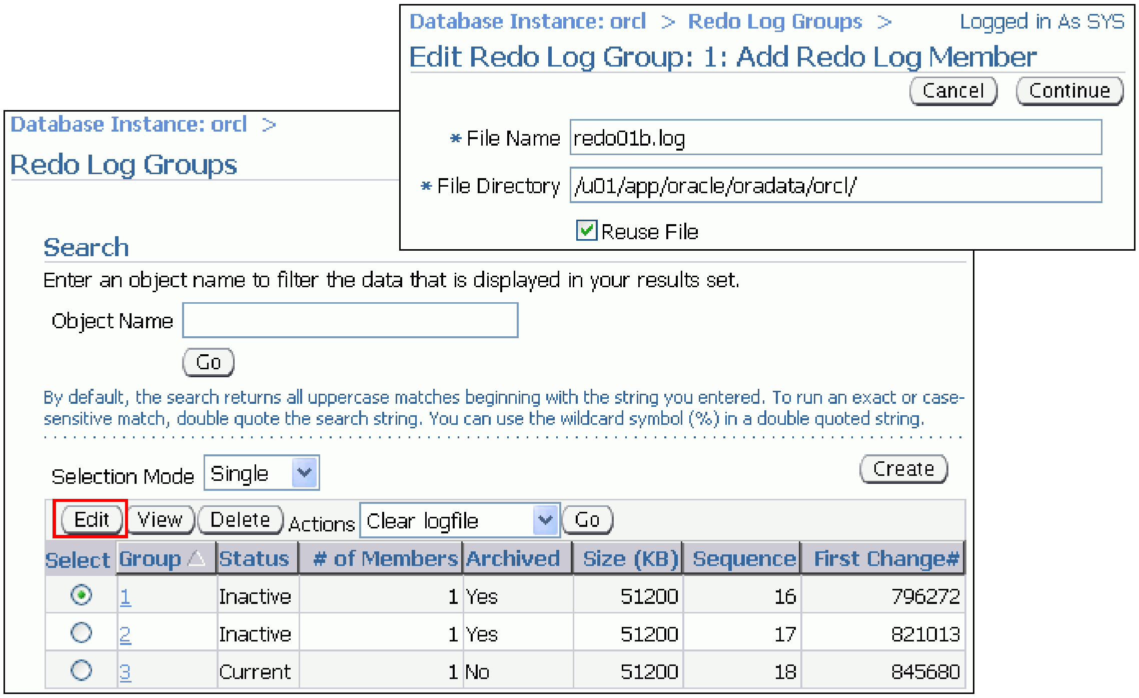
Redo Log Files

Redo log groups are made up of one or more redo log files. Each log file in a group is a duplicate of the others. Oracle Corporation recommends that redo log groups have at least two files per group, with the files distributed on separate disks or controllers so that no single equipment failure destroys an entire log group.

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

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

Multiplexing the Redo Log



Database Instance: orcl > Redo Log Groups > Logged in As SYS

Edit Redo Log Group: 1: Add Redo Log Member

(Cancel) (Continue)

* File Name: redo01b.log

* File Directory: /u01/app/oracle/oradata/orcl/

☑ Reuse File

Database Instance: orcl >

Redo Log Groups

Search

Enter an object name to filter the data that is displayed in your results set.

Object Name: (Go)

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode: Single (Create)

(Edit) (View) (Delete) Actions: Clear logfile (Go)

Select	Group △	Status	# of Members	Archived	Size (KB)	Sequence	First Change#
●	1	Inactive	1	Yes	51200	16	796272
○	2	Inactive	1	Yes	51200	17	821013
○	3	Current	1	No	51200	18	845680

14 - 24

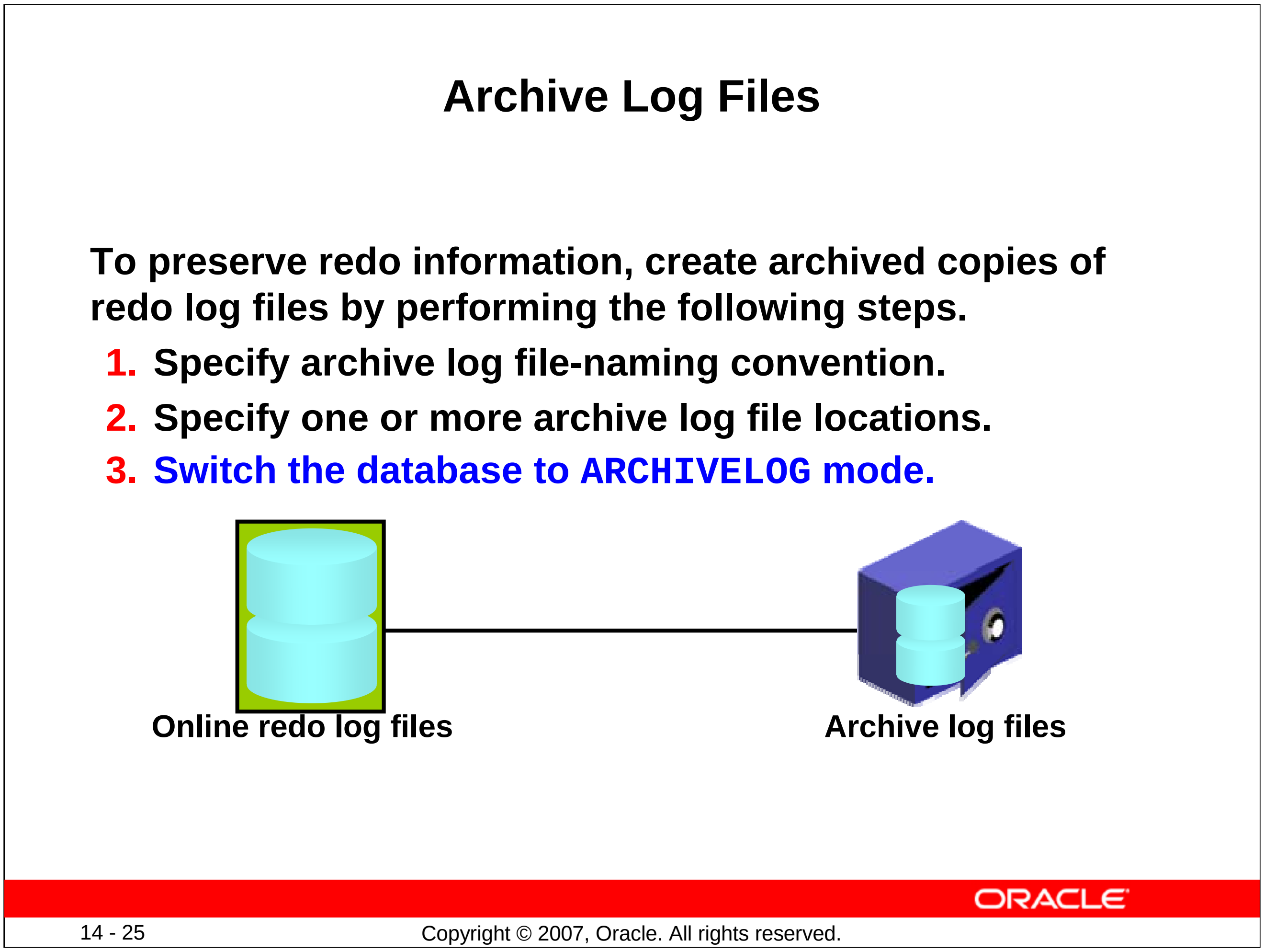
Multiplexing the Redo Log

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

1. Select Enterprise Manager > Server > Redo Log Groups.
2. Select a group and click the Edit button, or click the group number link.
   The Edit Redo Log Group page appears.
3. In the Redo Log Members region, click Add.
   The Add Redo Log Member page appears.
4. Enter the file name and the file directory. Click Continue.
   Note: It is recommended that you store members on separate drives to protect against total loss of the redo log entries in the event of a disk failure.

Repeat these steps for every existing group.

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



Archive Log Files

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

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

Online redo log files

Archive log files

Archive Log Files

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

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

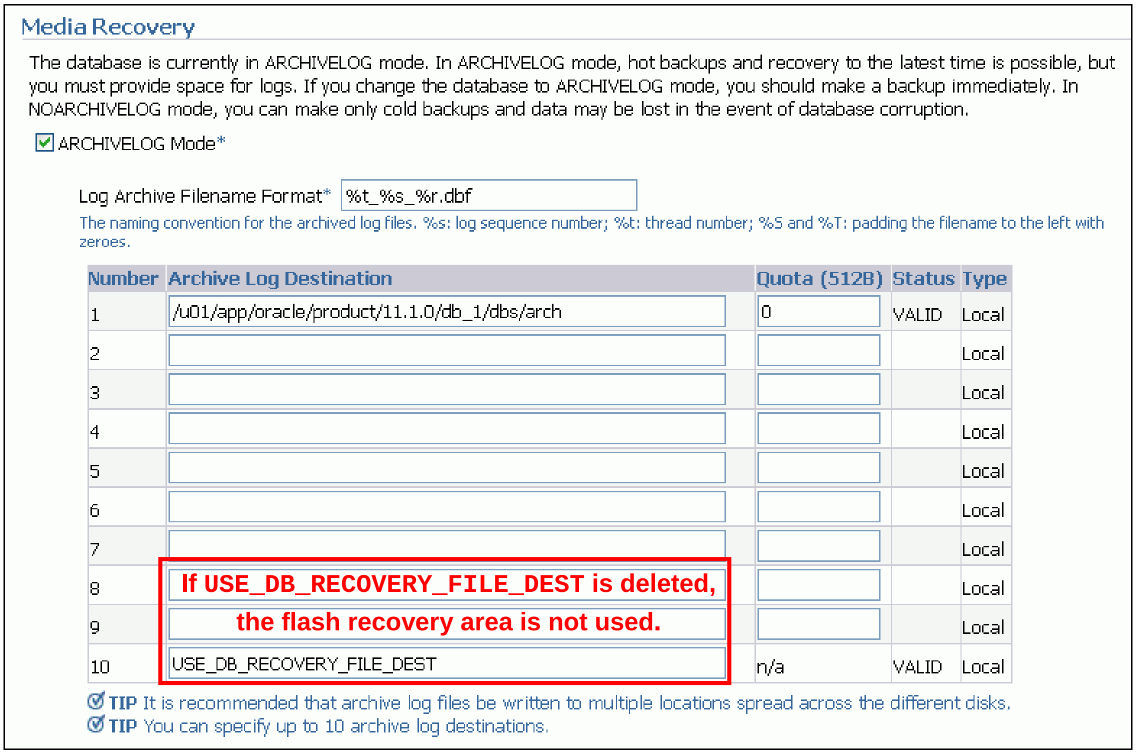
To facilitate the creation of archive log files:

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

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

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

Archive Log File: Naming and Destinations



Media Recovery

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

☑ ARCHIVELOG Mode*

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

The naming convention for the archived log files. %s: log sequence number; %t: thread number; %S and %T: padding the filename to the left with zeroes.

Number	Archive Log Destination	Quota (512B)	Status	Type
1	/u01/app/oracle/product/11.1.0/db_1/dbs/arch	0	VALID	Local
2				Local
3				Local
4				Local
5				Local
6				Local
7				Local
8				Local
9				Local
10	USE_DB_RECOVERY_FILE_DEST	n/a	VALID	Local

If USE_DB_RECOVERY_FILE_DEST is deleted, the flash recovery area is not used.

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

Archive Log File: Naming and Destinations

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

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

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

The format *must* include `%s`, `%t`, and `%r`. The use of `%d` is optional, but it must be included if multiple databases share the same archive log destination.

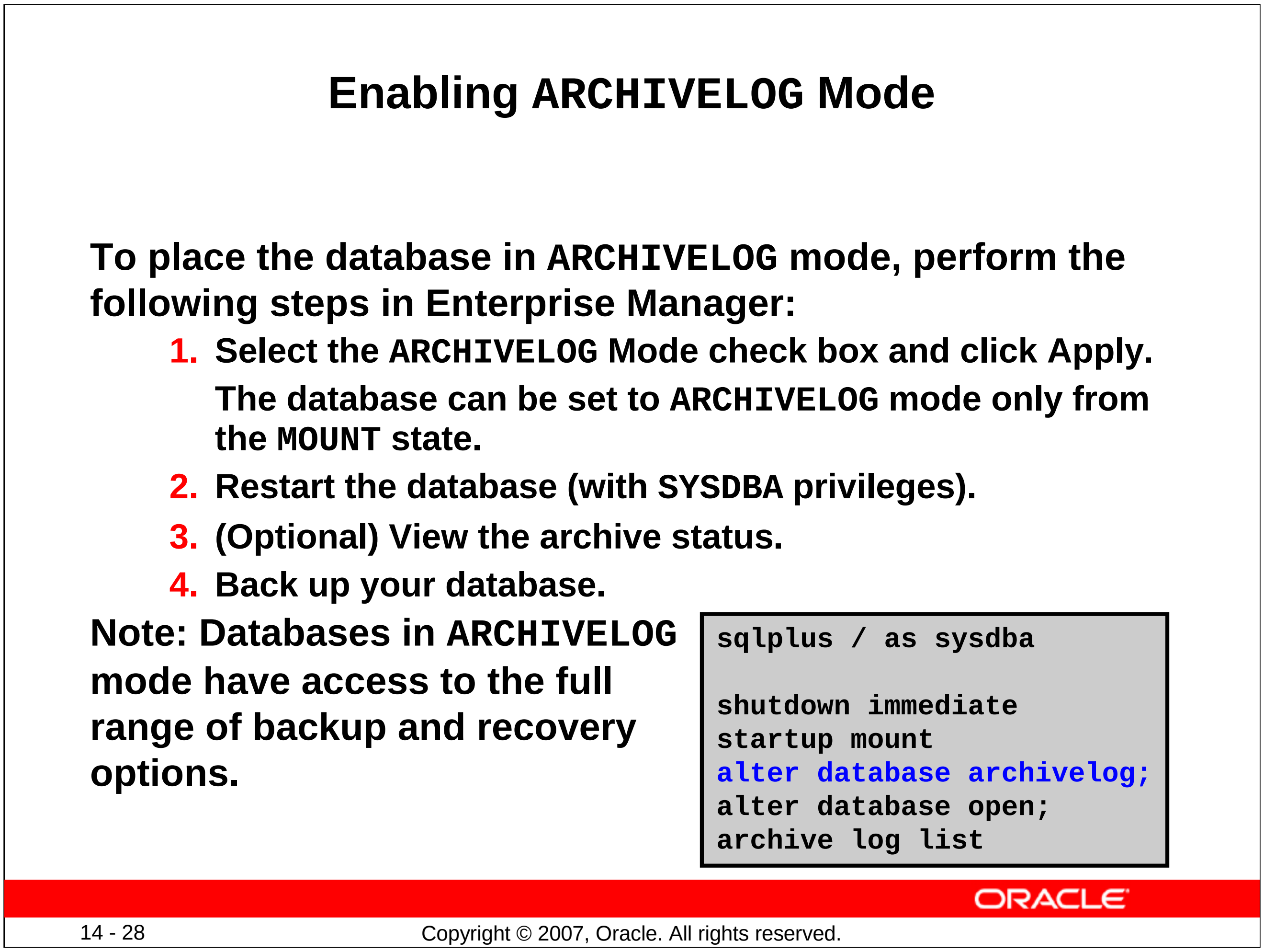
Archive log files can be written to as many as ten different destinations. Destinations may be local (a directory) or remote (an Oracle Net alias for a standby database).

Archive Log File: Naming and Destinations (continued)

The default destination (number 10) sends archive log files to a location determined by the DB_RECOVERY_FILE_DEST initialization parameter. DB_RECOVERY_FILE_DEST is also known as the RECOVERY AREA parameter. This destination is visible at the bottom of the Recovery Settings properties page as Flash Recovery Area Location.

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

To change recovery settings, you must be connected as SYSDBA or SYSOPER.



Enabling ARCHIVELOG Mode

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

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

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

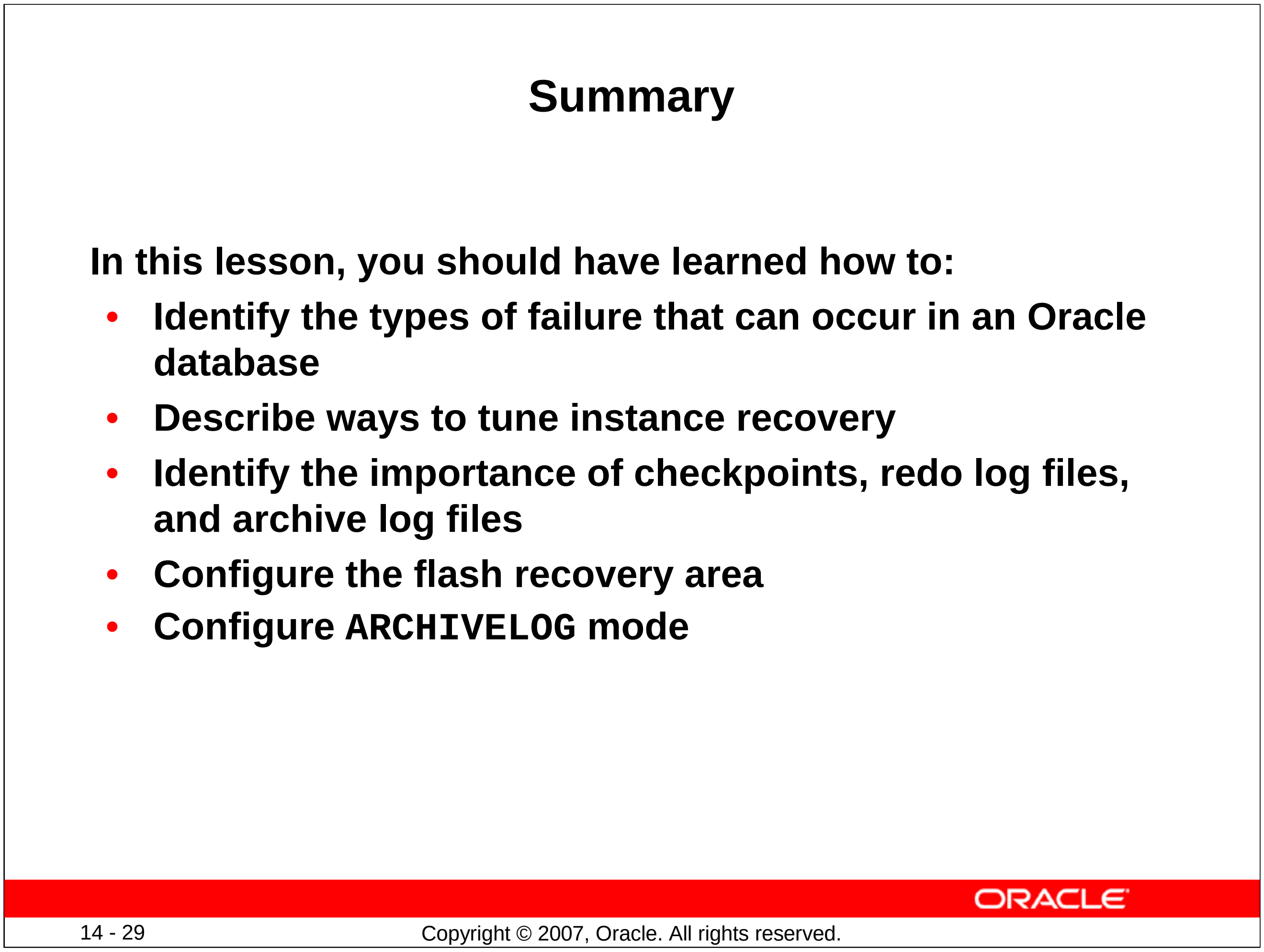
```
sqlplus / as sysdba

shutdown immediate
startup mount
alter database archivelog;
alter database open;
archive log list
```

Enabling ARCHIVELOG Mode

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

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

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

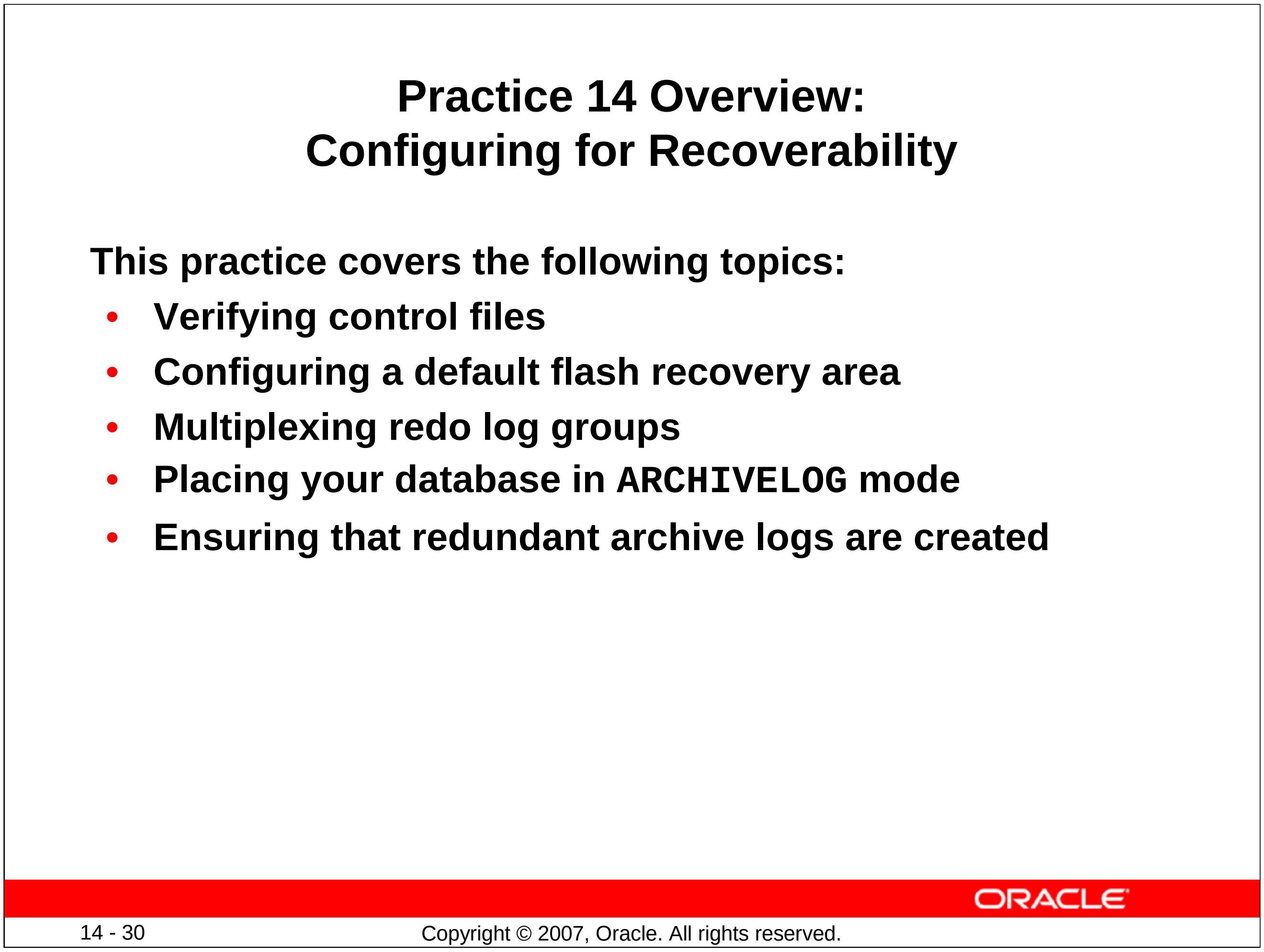


Summary

In this lesson, you should have learned how to:

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

14 - 29



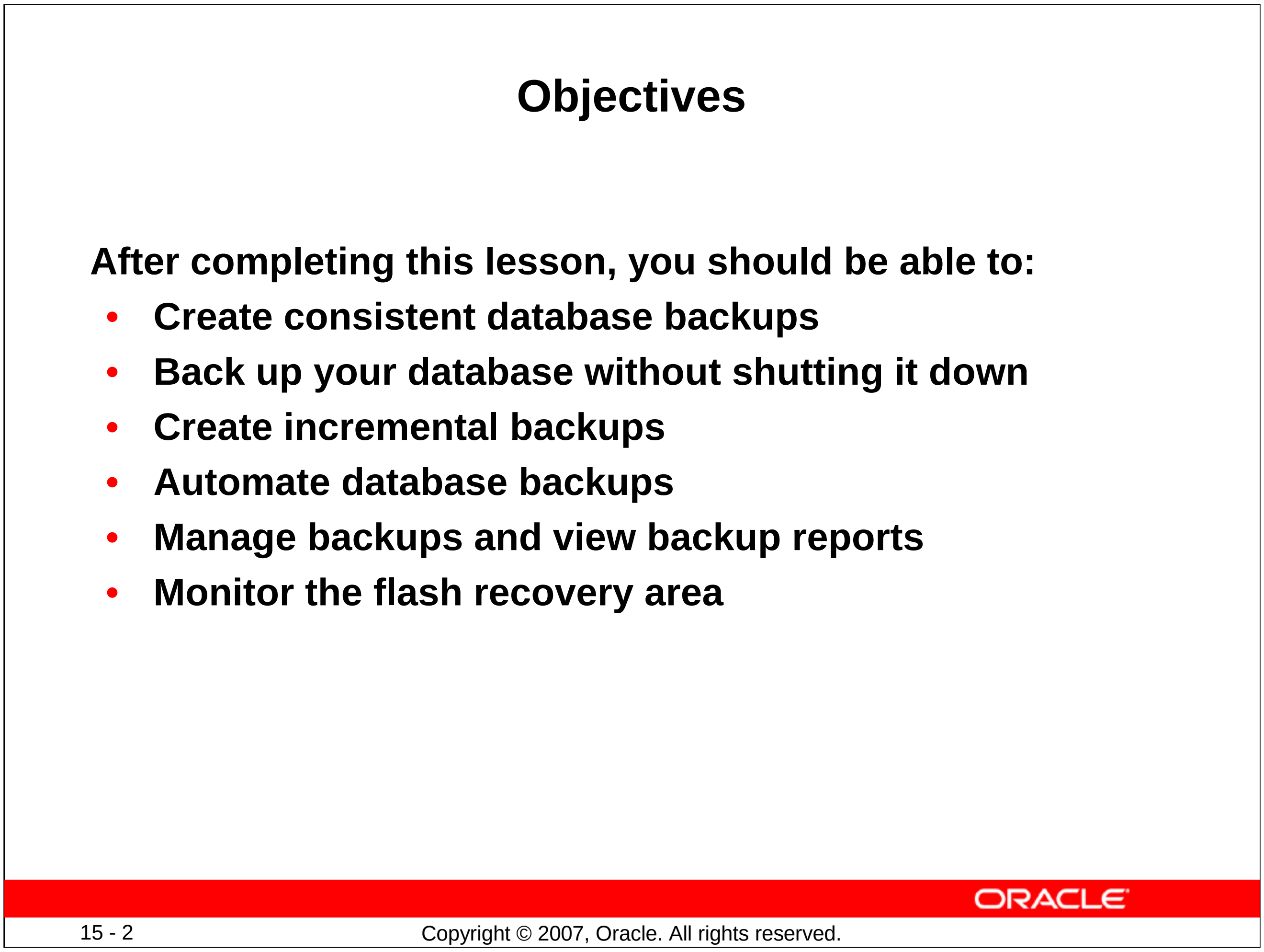
Practice 14 Overview: Configuring for Recoverability

This practice covers the following topics:

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

15

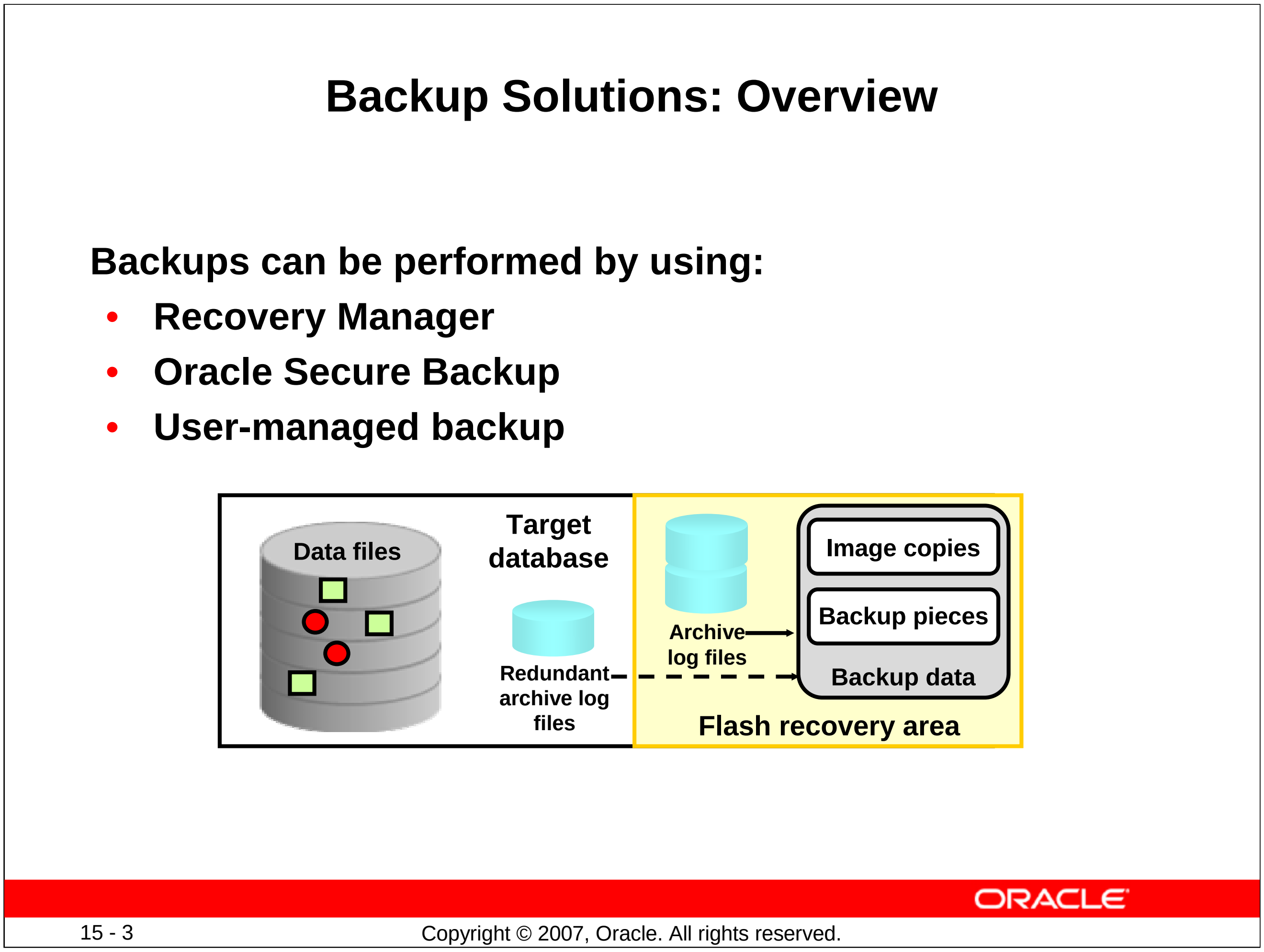
Performing Database Backups



Objectives

After completing this lesson, you should be able to:

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



Backup Solutions: Overview

Backups can be performed by using:

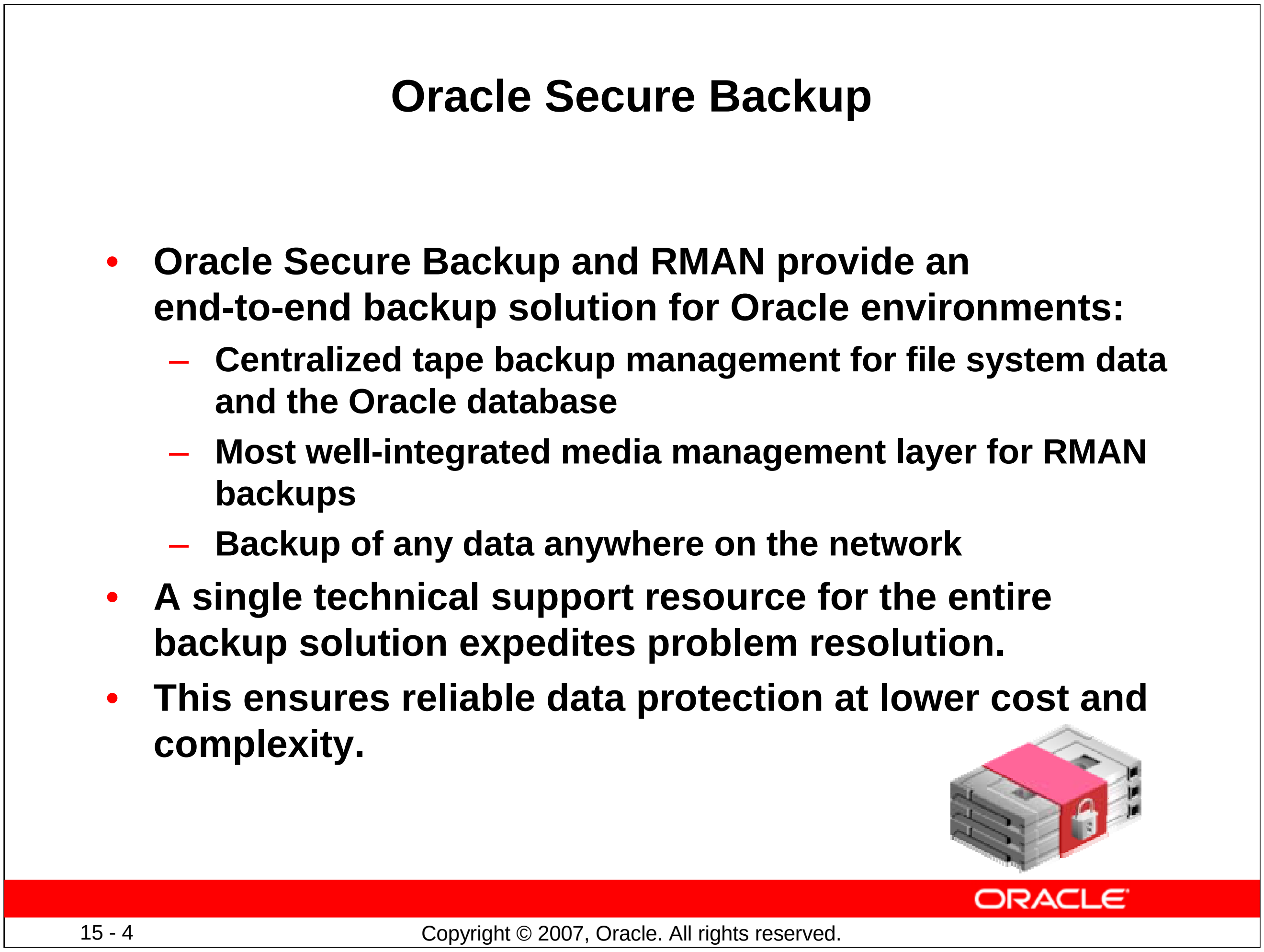
- Recovery Manager
- Oracle Secure Backup
- User-managed backup

Backup Solutions: Overview

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

Oracle Secure Backup complements existing functionality by adding backup to tape and network backup capabilities.

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



Oracle Secure Backup

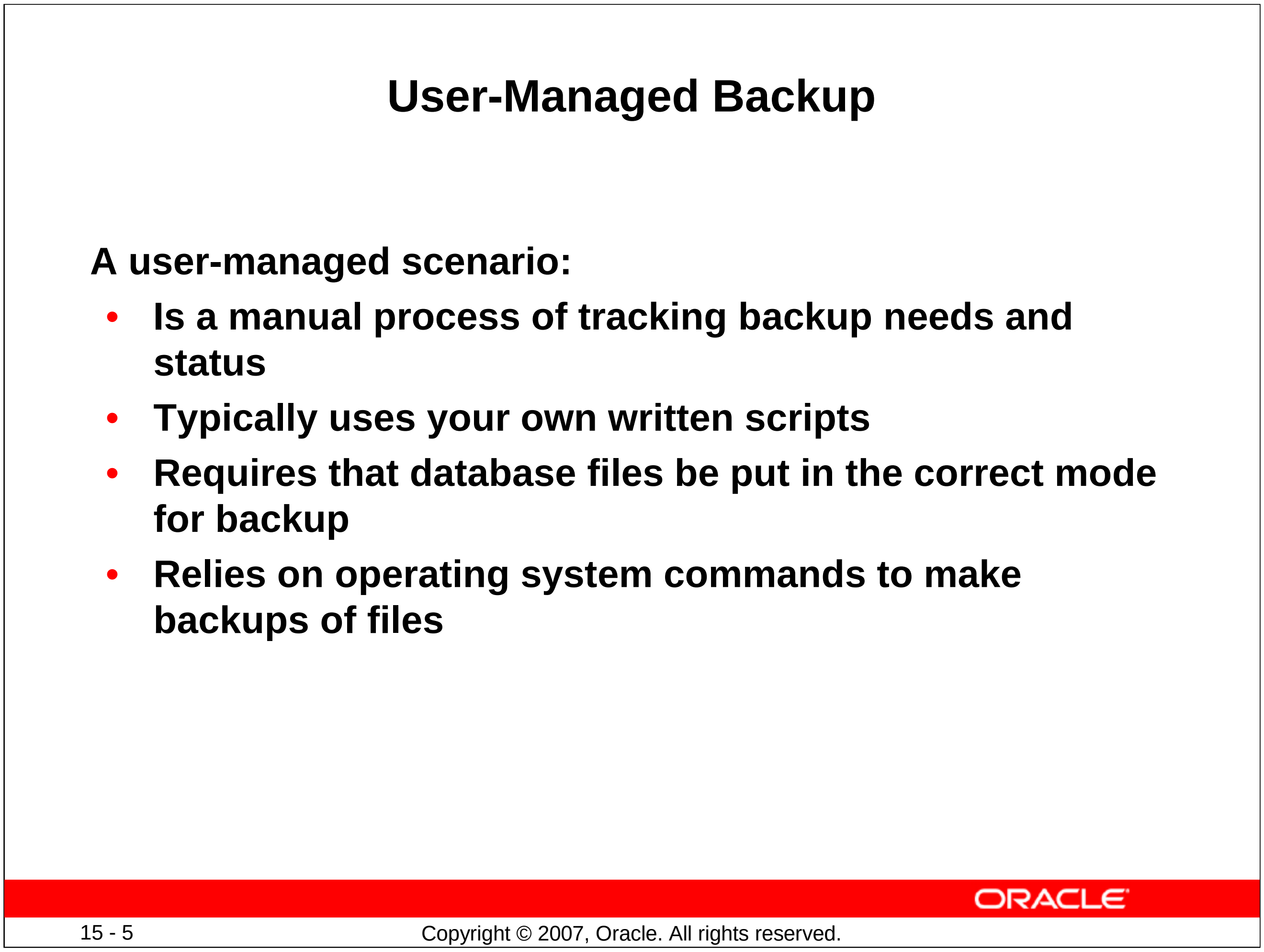
- Oracle Secure Backup and RMAN provide an end-to-end backup solution for Oracle environments:
  - Centralized tape backup management for file system data and the Oracle database
  - Most well-integrated media management layer for RMAN backups
  - Backup of any data anywhere on the network
- A single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

Oracle Secure Backup

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

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

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



User-Managed Backup

A user-managed scenario:

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

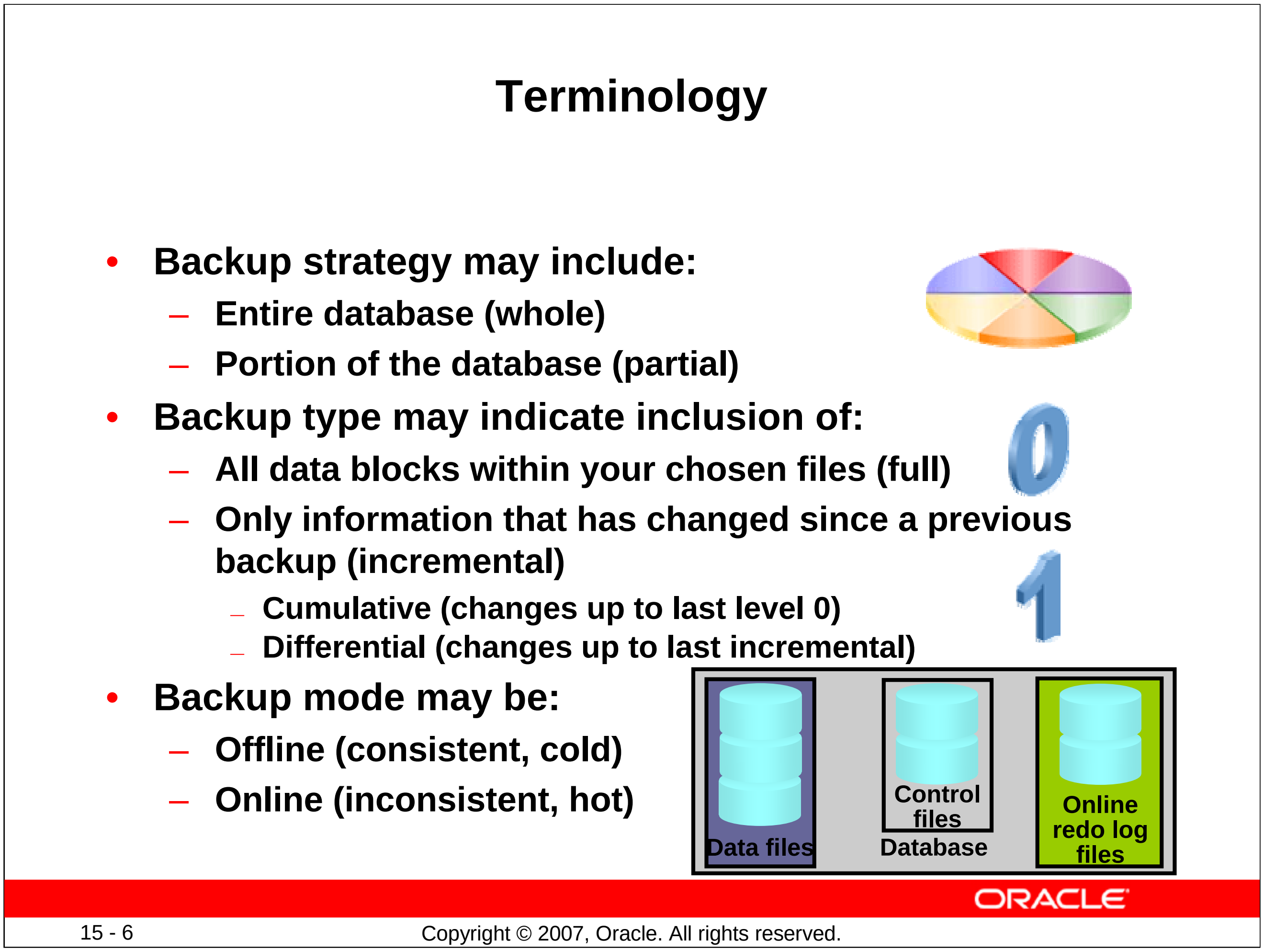
15 - 5

User-Managed Backup

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

Some of the actions that scripts must take:

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



Terminology

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

15 - 6

Terminology

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

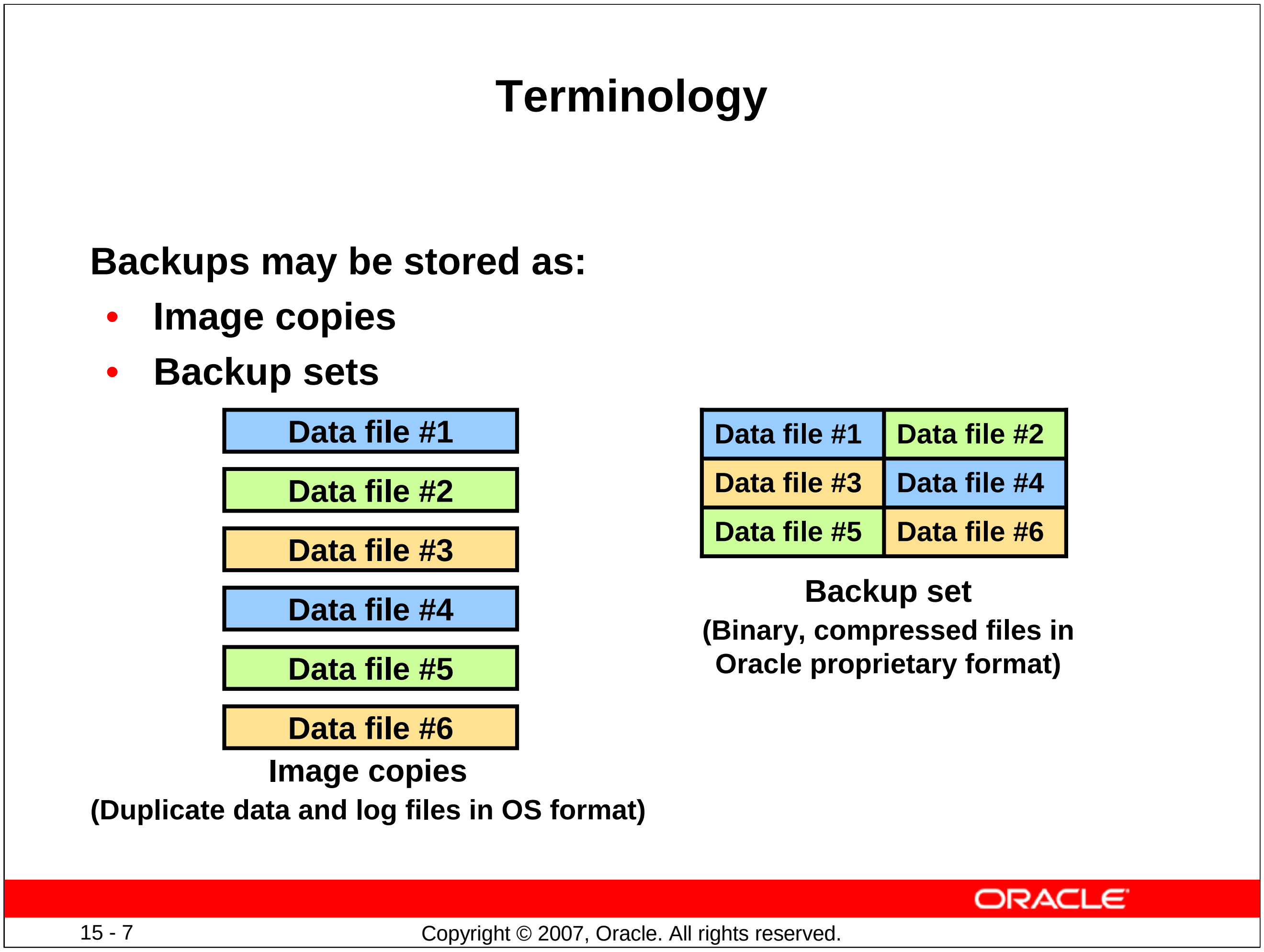
Partial database backup: May include zero or more tablespaces and zero or more data files; may or may not include a control file

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

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

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

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



Terminology

Backups may be stored as:

- Image copies
- Backup sets

Data file #1

Data file #2

Data file #3

Data file #4

Data file #5

Data file #6

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

Data file #1	Data file #2
Data file #3	Data file #4
Data file #5	Data file #6

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

Terminology (continued)

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

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

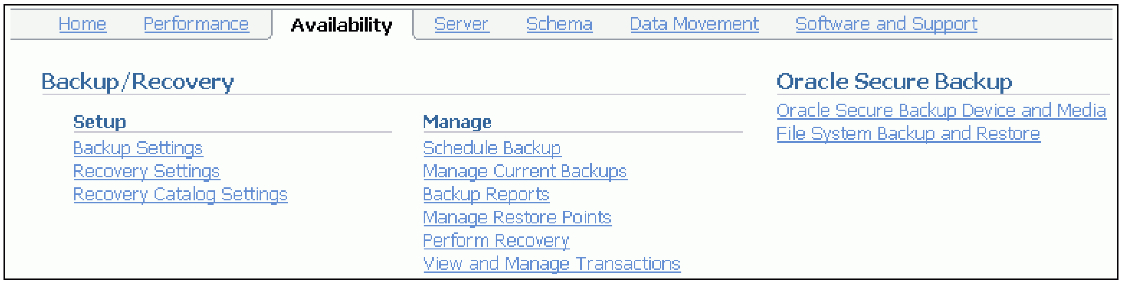
Image copies must be backed up to the disk. Backup sets can be sent to the disk or directly to the tape.

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

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

Recovery Manager (RMAN)

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



Home | Performance | **Availability** | Server | Schema | Data Movement | Software and Support

Backup/Recovery

Setup
- Backup Settings
- Recovery Settings
- Recovery Catalog Settings

Manage
- Schedule Backup
- Manage Current Backups
- Backup Reports
- Manage Restore Points
- Perform Recovery
- View and Manage Transactions

Oracle Secure Backup
- Oracle Secure Backup Device and Media
- File System Backup and Restore

Recovery Manager (RMAN)

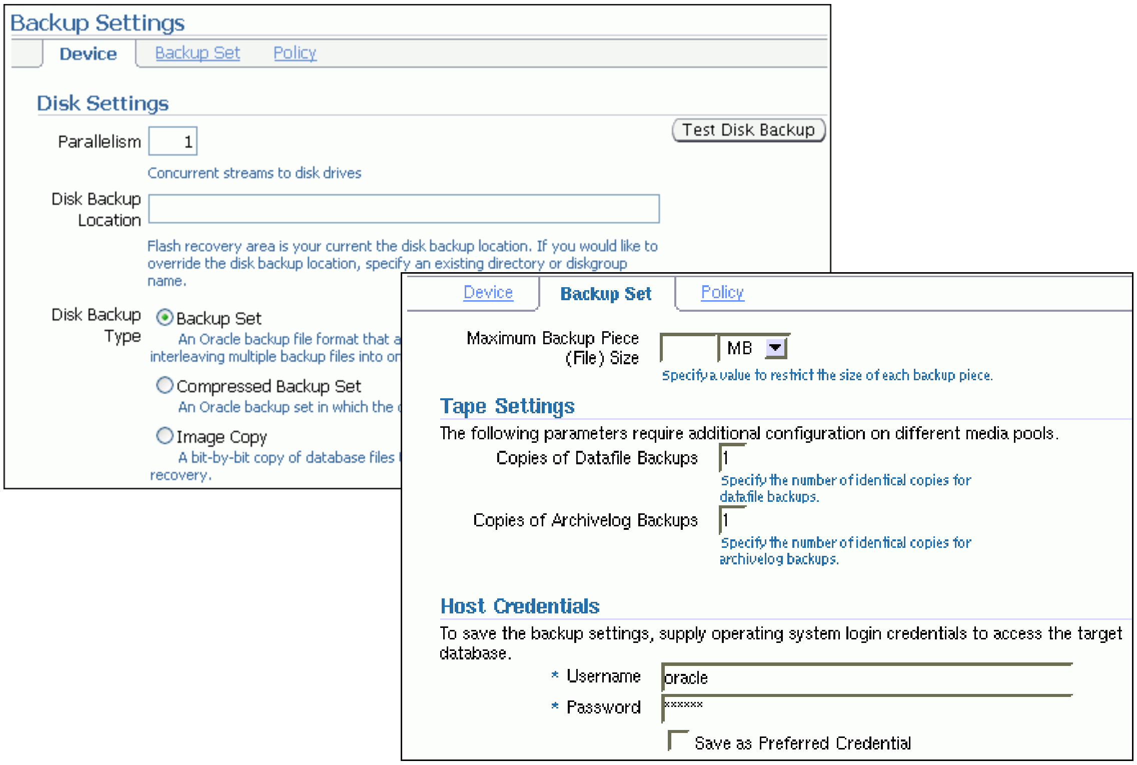
RMAN is the component of the Oracle database that is used to perform backup and recovery operations. It can make consistent and inconsistent backups, perform incremental and full backups, and back up either the whole database or a portion of it.

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

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

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

Configuring Backup Settings



Backup Settings

Device | Backup Set | Policy

Disk Settings

Test Disk Backup

Parallelism: 1

Concurrent streams to disk drives

Disk Backup Location:

Flash recovery area is your current the disk backup location. If you would like to override the disk backup location, specify an existing directory or diskgroup name.

Disk Backup Type:

● Backup Set
An Oracle backup file format that a
interleaving multiple backup files into or

○ Compressed Backup Set
An Oracle backup set in which the

○ Image Copy
A bit-by-bit copy of database files
recovery.

Device | Backup Set | Policy

Maximum Backup Piece (File) Size: MB

Specify a value to restrict the size of each backup piece.

Tape Settings

The following parameters require additional configuration on different media pools.

Copies of Datafile Backups: 1

Specify the number of identical copies for datafile backups.

Copies of Archivelog Backups: 1

Specify the number of identical copies for archivelog backups.

Host Credentials

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

* Username: oracle

* Password: ******

□ Save as Preferred Credential

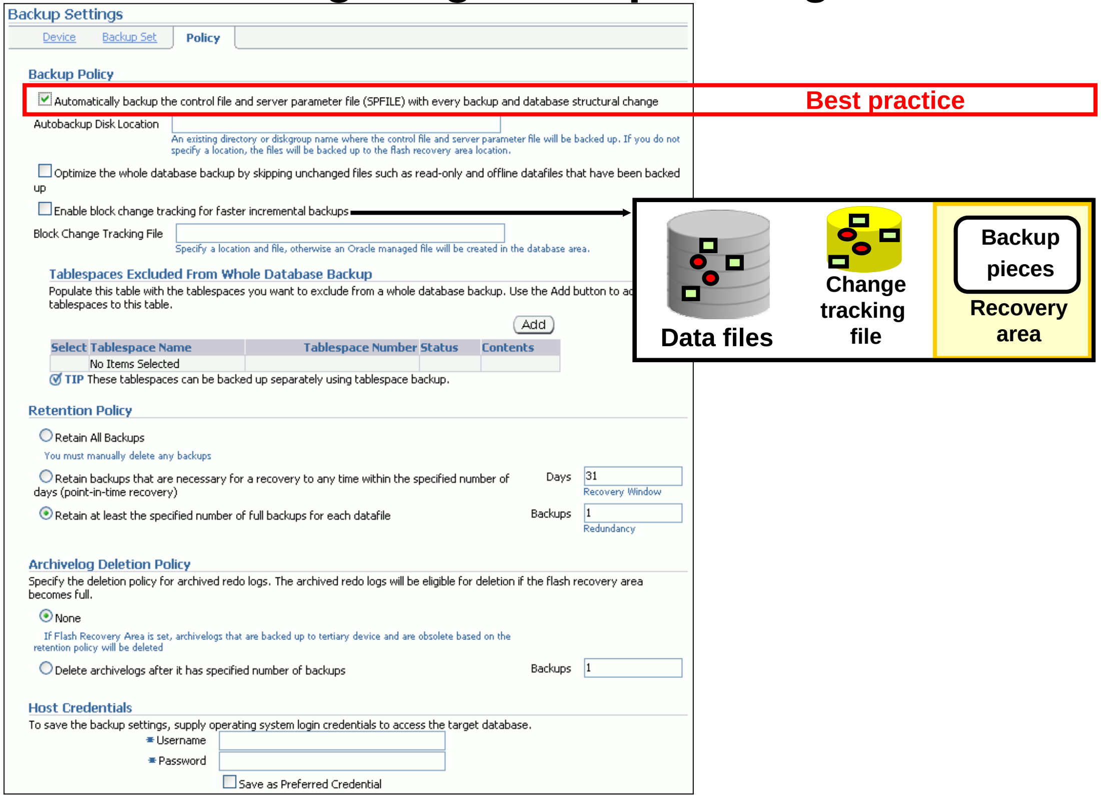
Configuring Backup Settings

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

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

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

Configuring Backup Settings



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.

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

☐ Enable block change tracking for faster incremental backups

Block Change Tracking File

Specify a location and file, otherwise an Oracle managed file will be created in the database area.

Tablespaces Excluded From Whole Database Backup

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

Select	Tablespace Name	Tablespace Number	Status	Contents
	No Items Selected			

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

Retention Policy

○ Retain All Backups

You must manually delete any backups

○ Retain backups that are necessary for a recovery to any time within the specified number of days (point-in-time recovery) — Days: 31 (Recovery Window)

● Retain at least the specified number of full backups for each datafile — Backups: 1 (Redundancy)

Archivelog Deletion Policy

Specify the deletion policy for archived redo logs. The archived redo logs will be eligible for deletion if the flash recovery area becomes full.

● None

If Flash Recovery Area is set, archivelogs that are backed up to tertiary device and are obsolete based on the retention policy will be deleted

○ Delete archivelogs after it has specified number of backups — Backups: 1

Host Credentials

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

* Username

* Password

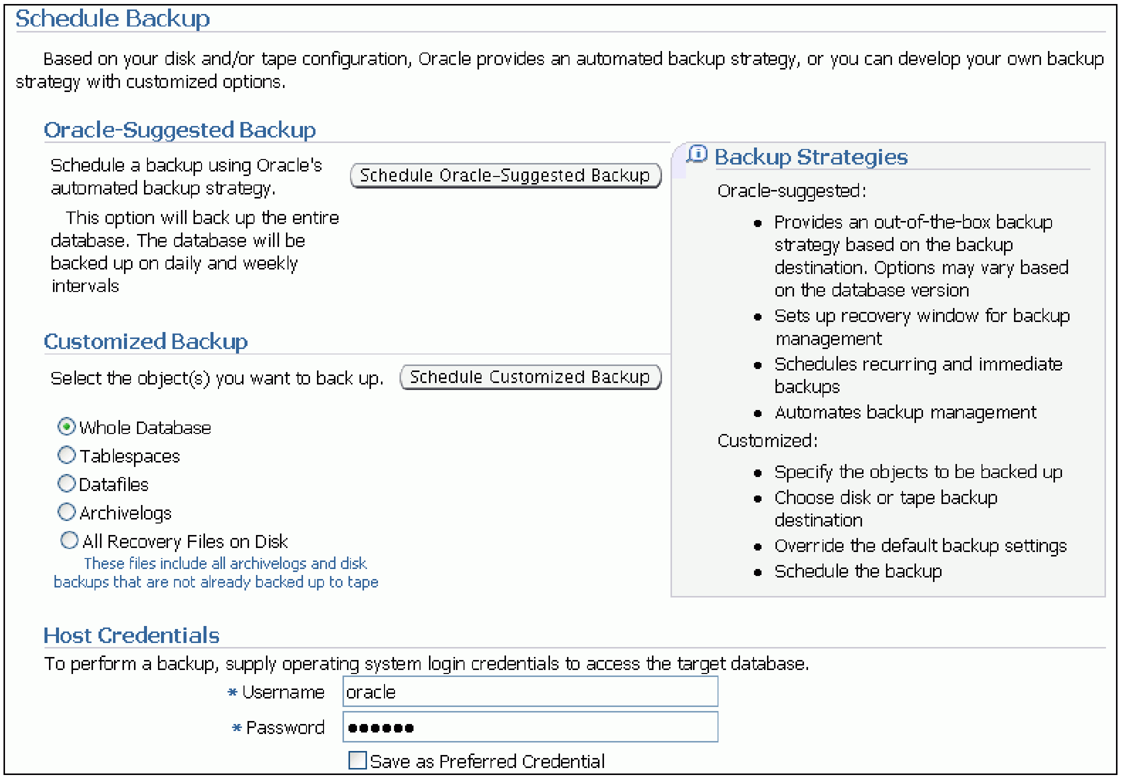
☐ Save as Preferred Credential

Configuring Backup Settings (continued)

Click the Policy tab to:

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

Scheduling Backups: Strategy



Schedule Backup

Based on your disk and/or tape configuration, Oracle provides an automated backup strategy, or you can develop your own backup strategy with customized options.

Oracle-Suggested Backup

Schedule a backup using Oracle's automated backup strategy.

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.

● Whole Database
○ Tablespaces
○ Datafiles
○ Archivelogs
○ All Recovery Files on Disk
These files include all archivelogs and disk backups that are not already backed up to tape

Backup Strategies

Oracle-suggested:

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

Customized:

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

Host Credentials

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

* Username: oracle

* Password: ••••••

□ Save as Preferred Credential

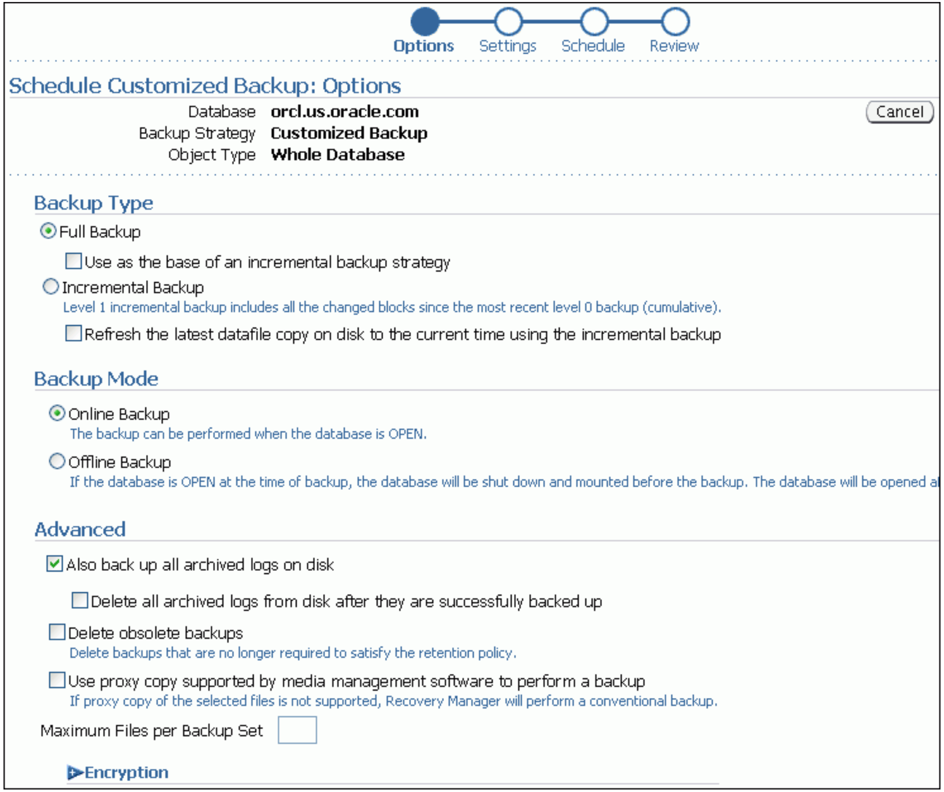
Scheduling Backups: Strategy

Select Enterprise Manager > Availability > Schedule Backup. Select either the Oracle-Suggested Backup strategy or your own customized strategy. The Oracle-Suggested Backup strategy makes a one-time whole-database backup, which is performed online. This is a baseline incremental level 0 backup. The automated backup strategy then schedules incremental level 1 backups for each successive day.

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

Both strategies enable you to set up encrypted backups.

Scheduling Backups: Options



Schedule Customized Backup: Options

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

Backup Type

● Full Backup

□ Use as the base of an incremental backup strategy

○ Incremental Backup
Level 1 incremental backup includes all the changed blocks since the most recent level 0 backup (cumulative).

□ Refresh the latest datafile copy on disk to the current time using the incremental backup

Backup Mode

● Online Backup
The backup can be performed when the database is OPEN.

○ Offline Backup
If the database is OPEN at the time of backup, the database will be shut down and mounted before the backup. The database will be opened a

Advanced

☑ Also back up all archived logs on disk

□ Delete all archived logs from disk after they are successfully backed up

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

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

Maximum Files per Backup Set □

▶ Encryption

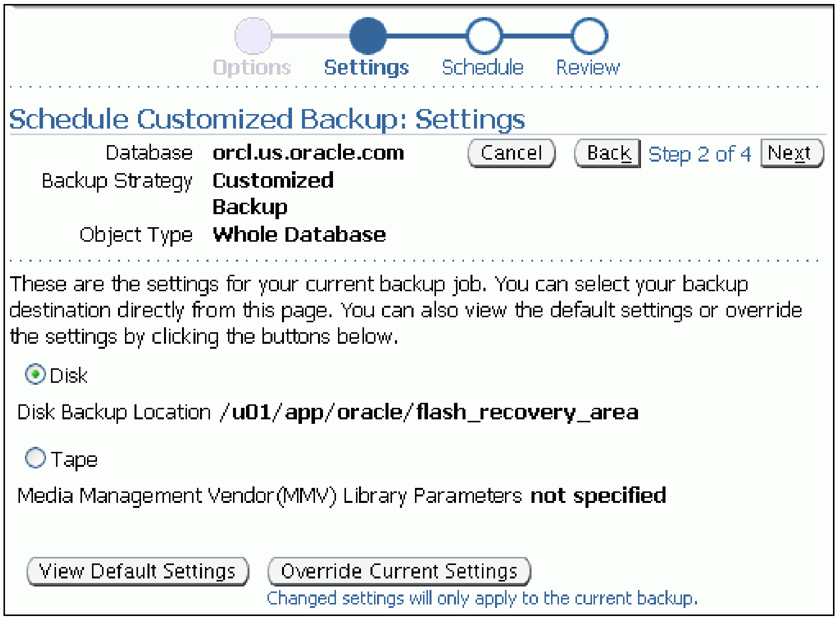
Scheduling Backups: Options

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

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

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

Scheduling Backups: Settings



Schedule Customized Backup: Settings

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

These are the settings for your current backup job. You can select your backup destination directly from this page. You can also view the default settings or override the settings by clicking the buttons below.

● Disk

Disk Backup Location **/u01/app/oracle/flash_recovery_area**

○ Tape

Media Management Vendor(MMV) Library Parameters **not specified**

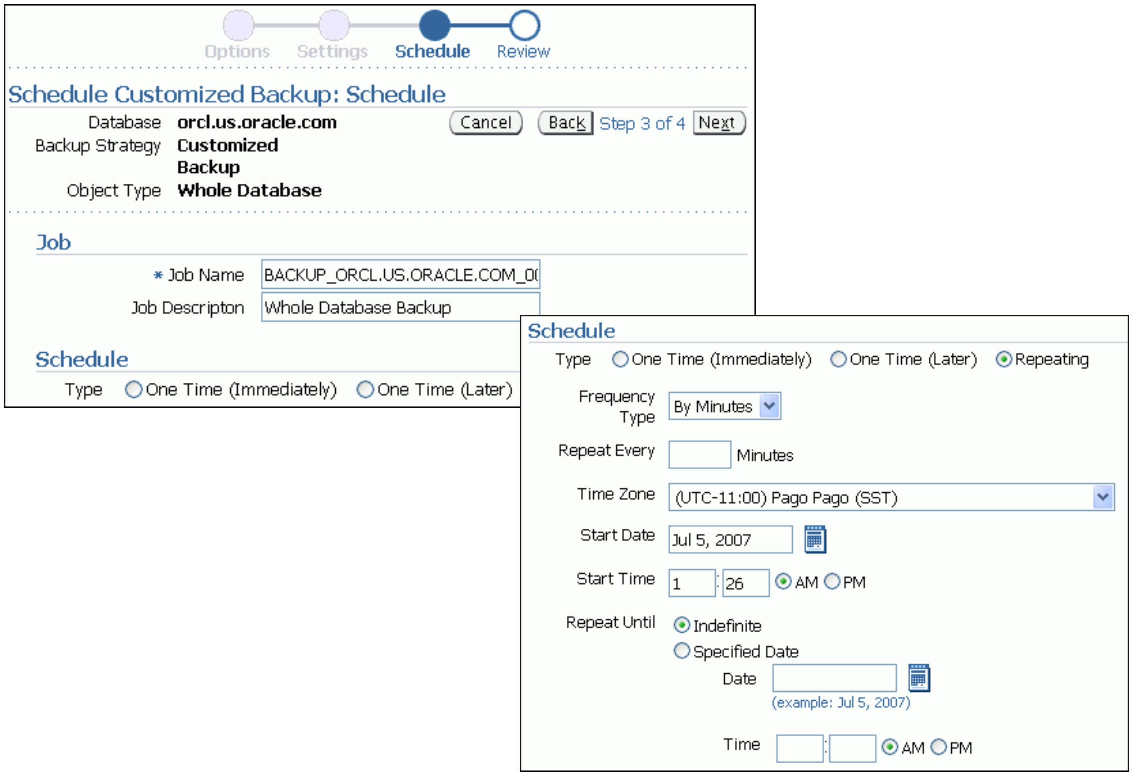
Changed settings will only apply to the current backup.

Scheduling Backups: Settings

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

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

Scheduling Backups: Schedule



Schedule Customized Backup: Schedule

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

Job

* Job Name	BACKUP_ORCL.US.ORACLE.COM_0(
Job Descripton	Whole Database Backup

Schedule

Type ○ One Time (Immediately) ○ One Time (Later)

Schedule

Type	○ One Time (Immediately) ○ One Time (Later) ● Repeating
Frequency Type	By Minutes
Repeat Every	Minutes
Time Zone	(UTC-11:00) Pago Pago (SST)
Start Date	Jul 5, 2007
Start Time	1 : 26 ● AM ○ PM
Repeat Until	● Indefinite ○ Specified Date Date (example: Jul 5, 2007) Time : ● AM ○ PM

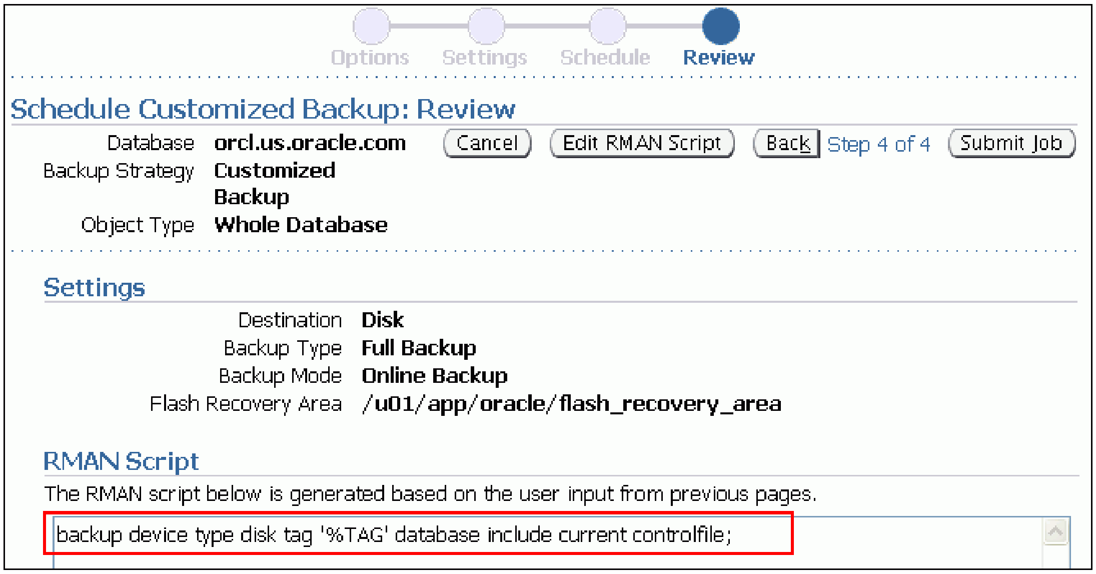
Scheduling Backups: Schedule

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

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

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

Scheduling Backups: Review



Schedule Customized Backup: Review

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

Settings

Destination	Disk
Backup Type	Full Backup
Backup Mode	Online Backup
Flash Recovery Area	/u01/app/oracle/flash_recovery_area

RMAN Script

The RMAN script below is generated based on the user input from previous pages.

```
backup device type disk tag '%TAG' database include current controlfile;
```

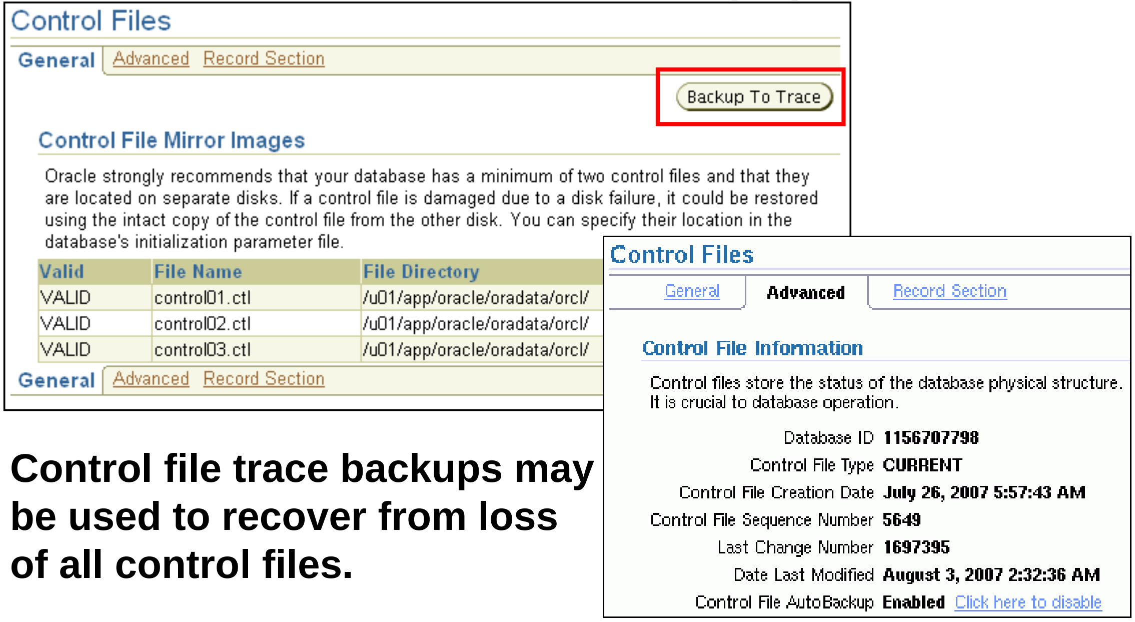
Scheduling Backups: Review

RMAN uses its own command syntax and scripting language.

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

Backing Up the Control File to a Trace File

Control files have an additional backup option.



Control Files

Control File Mirror Images

Oracle strongly recommends that your database has a minimum of two control files and that they are located on separate disks. If a control file is damaged due to a disk failure, it could be restored using the intact copy of the control file from the other disk. You can specify their location in the database's initialization parameter file.

Valid	File Name	File Directory
VALID	control01.ctl	/u01/app/oracle/oradata/orcl/
VALID	control02.ctl	/u01/app/oracle/oradata/orcl/
VALID	control03.ctl	/u01/app/oracle/oradata/orcl/

Control Files

Control File Information

Control files store the status of the database physical structure. It is crucial to database operation.

Database ID	1156707798
Control File Type	CURRENT
Control File Creation Date	July 26, 2007 5:57:43 AM
Control File Sequence Number	5649
Last Change Number	1697395
Date Last Modified	August 3, 2007 2:32:36 AM
Control File AutoBackup	Enabled Click here to disable

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

Backing Up the Control File to a Trace File

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

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

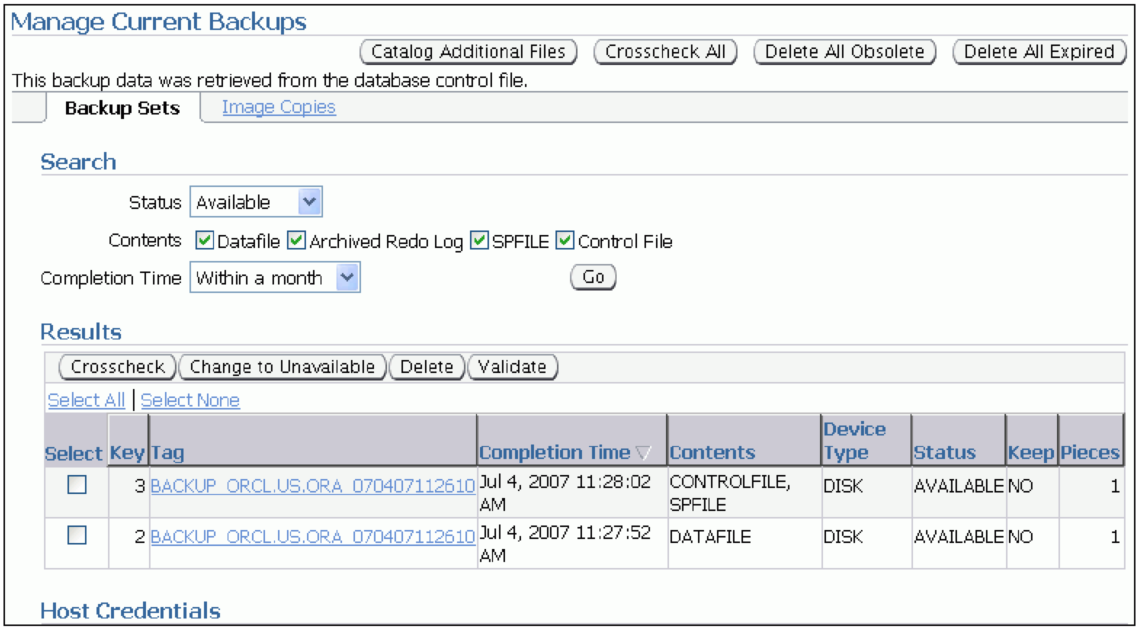
Trace copies of the control file can be created by using Enterprise Manager (as shown in the slide) or with the following SQL command:

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

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

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

Managing Backups



Manage Current Backups

This backup data was retrieved from the database control file.

Search

Status: Available

Contents: ☑ Datafile ☑ Archived Redo Log ☑ SPFILE ☑ Control File

Completion Time: Within a month

Results

Select	Key	Tag	Completion Time ▽	Contents	Device Type	Status	Keep	Pieces
□	3	BACKUP_ORCL.US.ORA_070407112610	Jul 4, 2007 11:28:02 AM	CONTROLFILE, SPFILE	DISK	AVAILABLE	NO	1
□	2	BACKUP_ORCL.US.ORA_070407112610	Jul 4, 2007 11:27:52 AM	DATAFILE	DISK	AVAILABLE	NO	1

Host Credentials

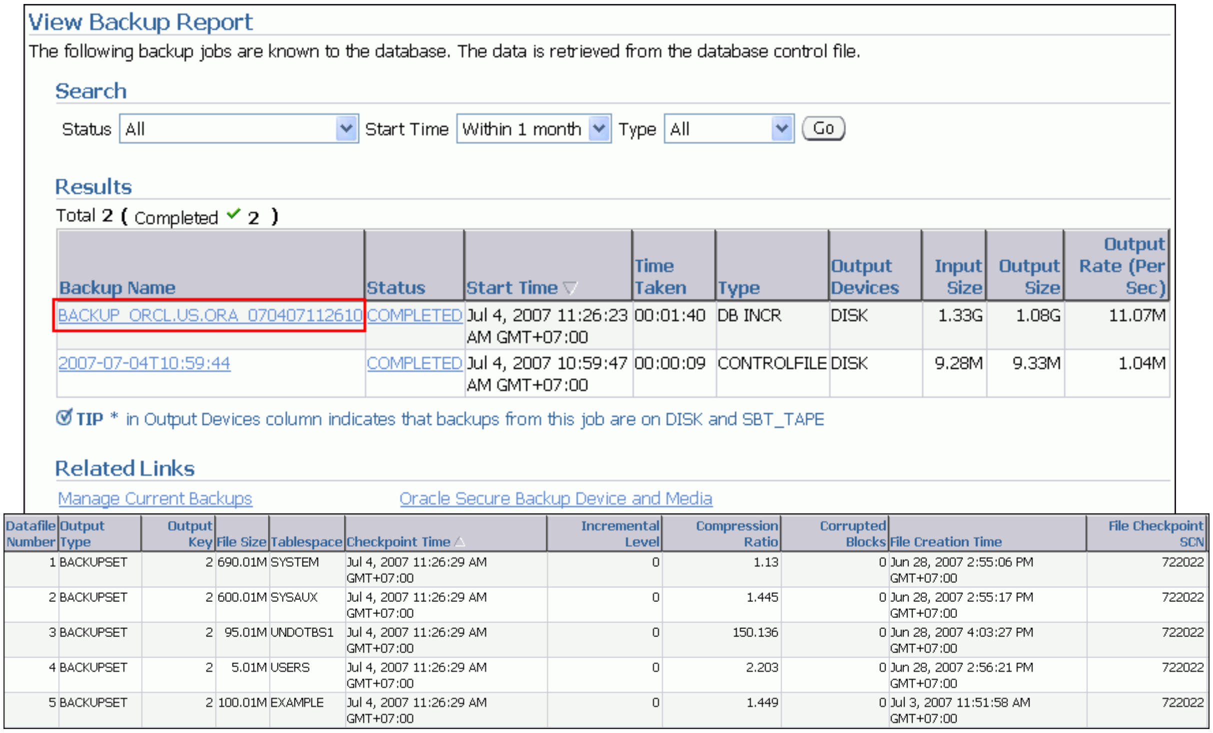
Managing Backups

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

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

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

Viewing Backup Reports



View Backup Report

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

Search

Status All Start Time Within 1 month Type All Go

Results

Total 2 (Completed ✓ 2)

Backup Name	Status	Start Time ▽	Time Taken	Type	Output Devices	Input Size	Output Size	Output Rate (Per Sec)
BACKUP_ORCL.US.ORA_070407112610	COMPLETED	Jul 4, 2007 11:26:23 AM GMT+07:00	00:01:40	DB INCR	DISK	1.33G	1.08G	11.07M
2007-07-04T10:59:44	COMPLETED	Jul 4, 2007 10:59:47 AM GMT+07:00	00:00:09	CONTROLFILE	DISK	9.28M	9.33M	1.04M

TIP * in Output Devices column indicates that backups from this job are on DISK and SBT_TAPE

Related Links

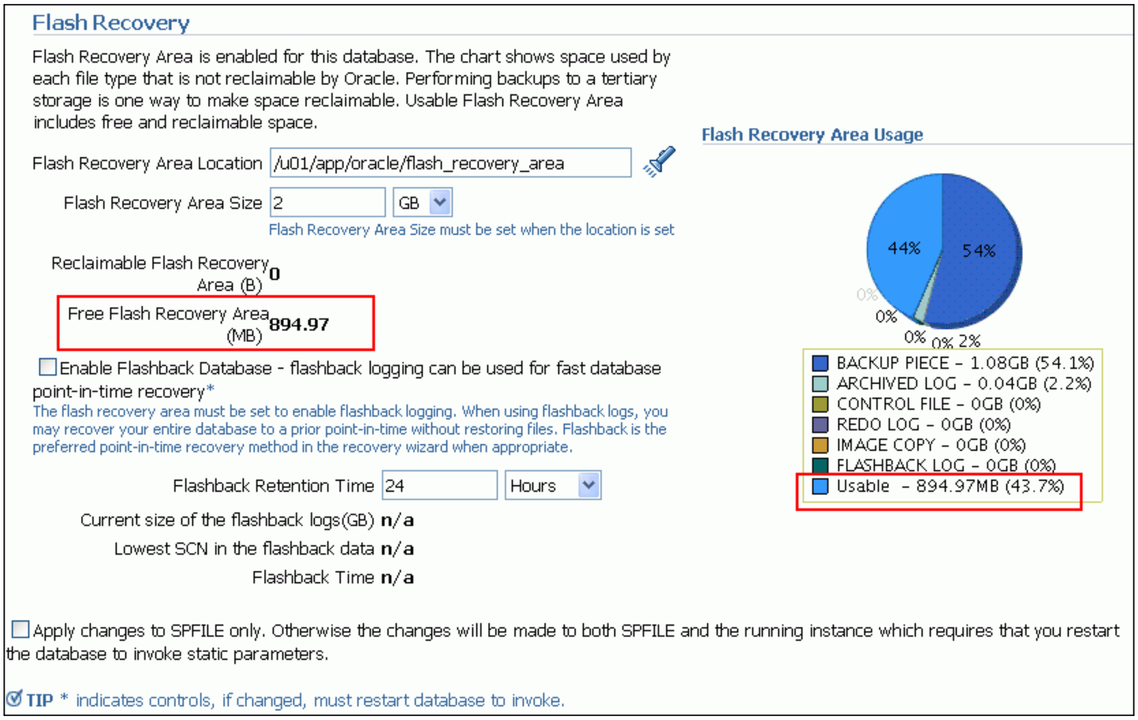
Manage Current Backups Oracle Secure Backup Device and Media

Datafile Number	Output Type	Output Key	File Size	Tablespace	Checkpoint Time △	Incremental Level	Compression Ratio	Corrupted Blocks	File Creation Time	File Checkpoint SCN
1	BACKUPSET	2	690.01M	SYSTEM	Jul 4, 2007 11:26:29 AM GMT+07:00	0	1.13	0	Jun 28, 2007 2:55:06 PM GMT+07:00	722022
2	BACKUPSET	2	600.01M	SYSAUX	Jul 4, 2007 11:26:29 AM GMT+07:00	0	1.445	0	Jun 28, 2007 2:55:17 PM GMT+07:00	722022
3	BACKUPSET	2	95.01M	UNDOTBS1	Jul 4, 2007 11:26:29 AM GMT+07:00	0	150.136	0	Jun 28, 2007 4:03:27 PM GMT+07:00	722022
4	BACKUPSET	2	5.01M	USERS	Jul 4, 2007 11:26:29 AM GMT+07:00	0	2.203	0	Jun 28, 2007 2:56:21 PM GMT+07:00	722022
5	BACKUPSET	2	100.01M	EXAMPLE	Jul 4, 2007 11:26:29 AM GMT+07:00	0	1.449	0	Jul 3, 2007 11:51:58 AM GMT+07:00	722022

Viewing Backup Reports

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

Monitoring the Flash Recovery Area



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 /u01/app/oracle/flash_recovery_area

Flash Recovery Area Size 2 GB

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

Reclaimable Flash Recovery Area (B) 0

Free Flash Recovery Area (MB) 894.97

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

The flash recovery 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 24 Hours

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

Lowest SCN in the flashback data n/a

Flashback Time n/a

Flash Recovery Area Usage

- 🟦 BACKUP PIECE - 1.08GB (54.1%)
- 🟩 ARCHIVED LOG - 0.04GB (2.2%)
- 🟨 CONTROL FILE - 0GB (0%)
- 🟪 REDO LOG - 0GB (0%)
- 🟧 IMAGE COPY - 0GB (0%)
- 🟩 FLASHBACK LOG - 0GB (0%)
- 🟦 Usable - 894.97MB (43.7%)

□ 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.

TIP * indicates controls, if changed, must restart database to invoke.

ORACLE

Monitoring the Flash Recovery Area

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

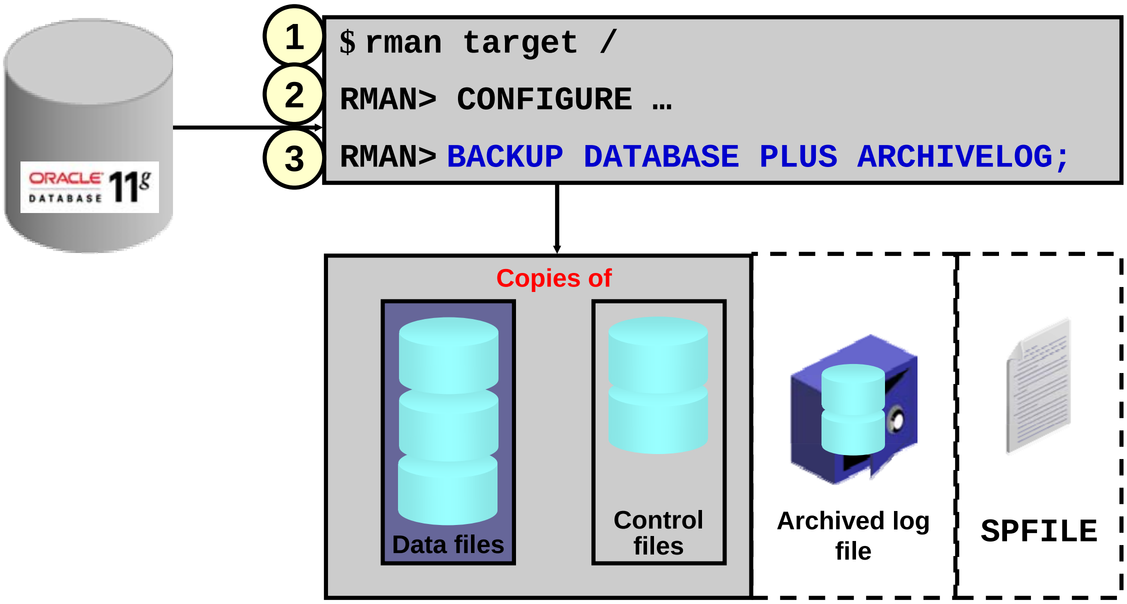
Select Enterprise Manager > Availability > Recovery Settings. On this page, you can:

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

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

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

Using the RMAN Command Line



```
1 $ rman target /
2 RMAN> CONFIGURE …
3 RMAN> BACKUP DATABASE PLUS ARCHIVELOG;
```

Using the RMAN Command Line

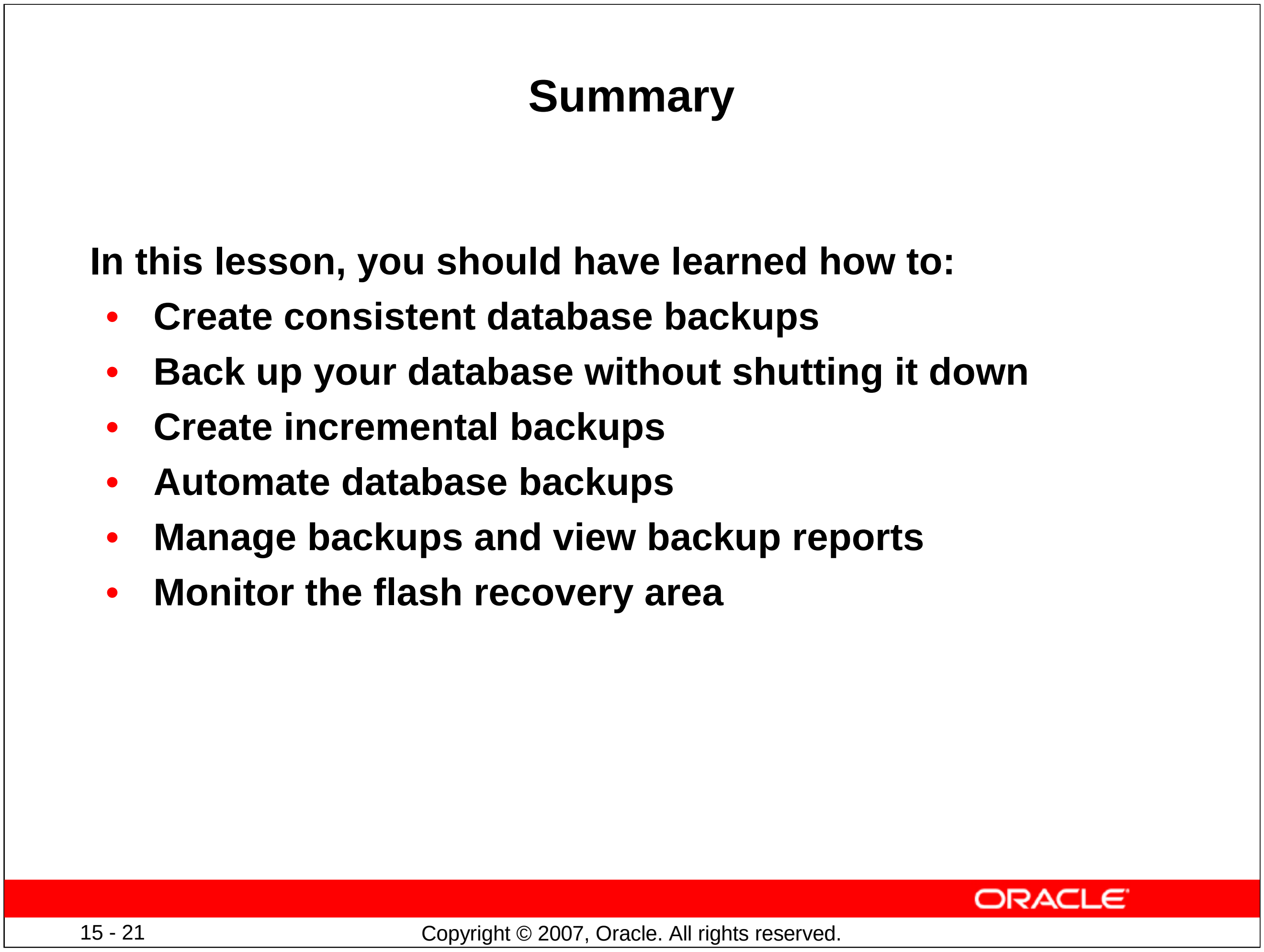
1. In a terminal session, start RMAN and connect to the target database.
2. Execute configuration commands:
   - CONFIGURE DEFAULT DEVICE TYPE TO disk;
   - CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO COPY;
   - CONFIGURE CONTROLFILE AUTOBACKUP ON;
3. A whole database backup is a copy of all data files and the control file. You can optionally include the server parameter file (SPFILE) and archived redo log files. Using RMAN to make an image copy of all the database files simply requires mounting or opening the database, starting RMAN, and entering the BACKUP command shown in the slide.

   Optionally, you can supply the DELETE INPUT option when backing up archive log files. That causes RMAN to remove the archive log files after backing them up. This is useful especially if you are not using a 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 the following:

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

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

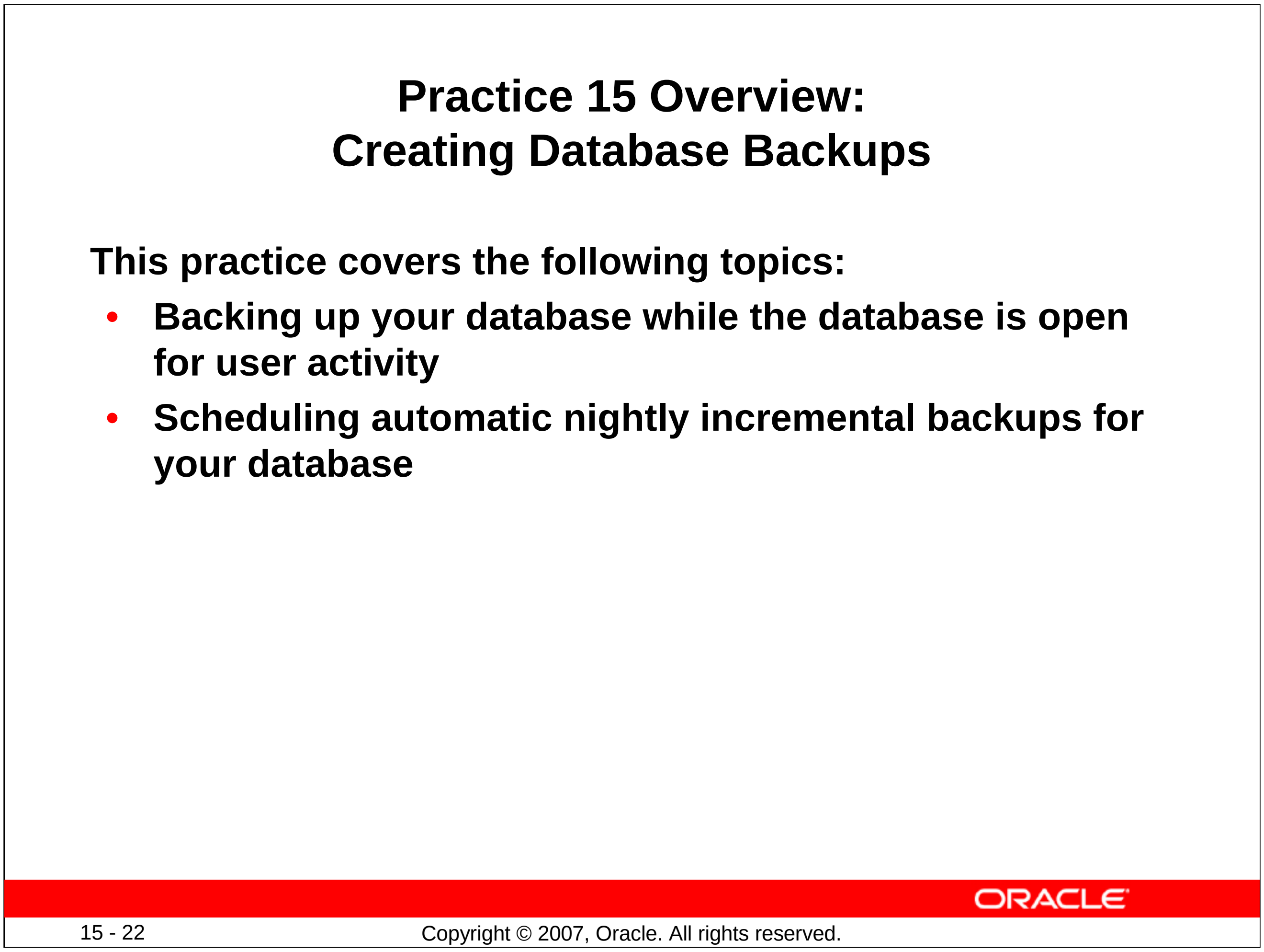
   ```
   RMAN> BACKUP COPY OF DATABASE;
   ```



Summary

In this lesson, you should have learned how to:

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



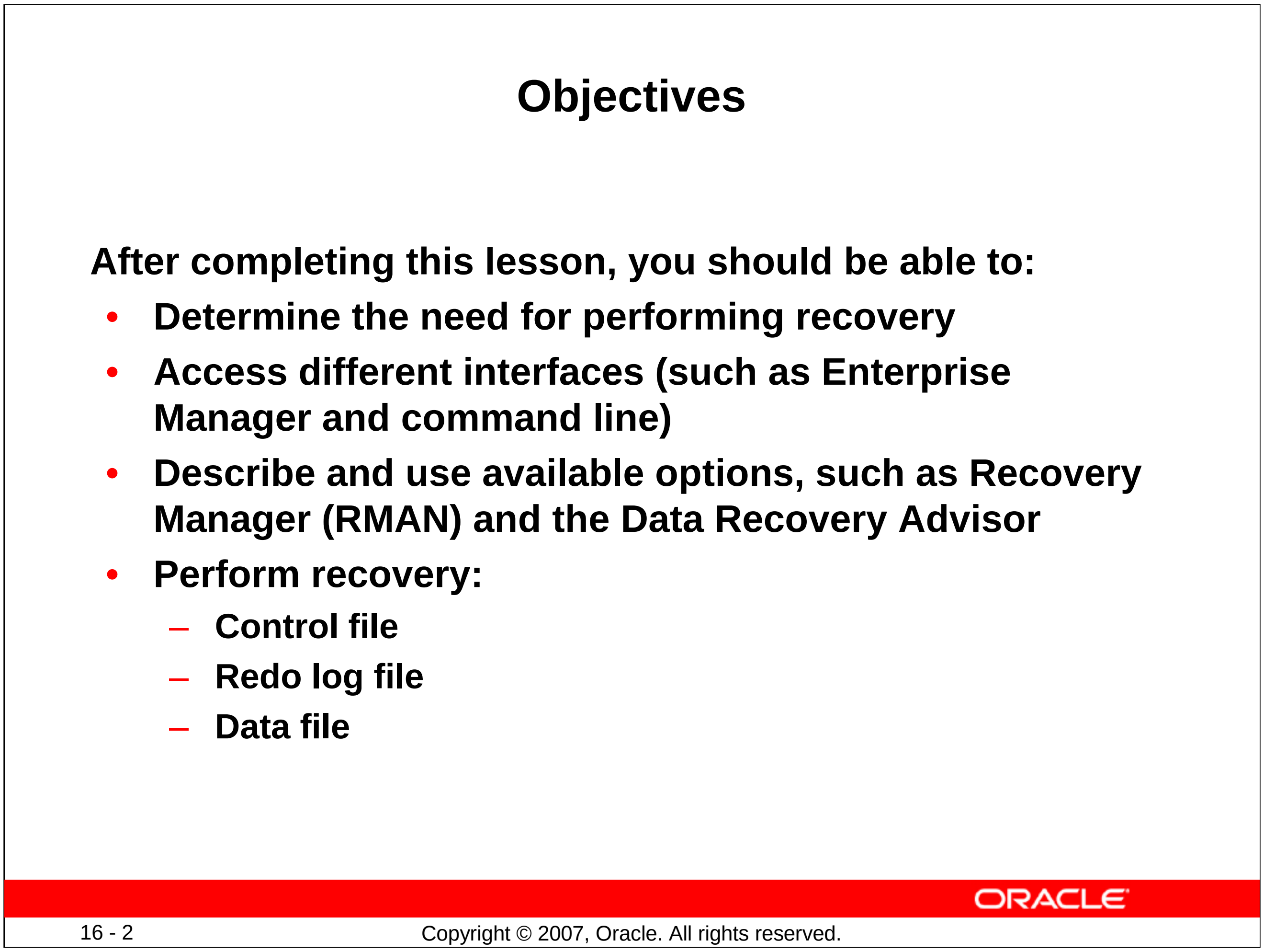
Practice 15 Overview: Creating Database Backups

This practice covers the following topics:

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

16

Performing Database Recovery



Objectives

After completing this lesson, you should be able to:

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

Opening a Database (continued)

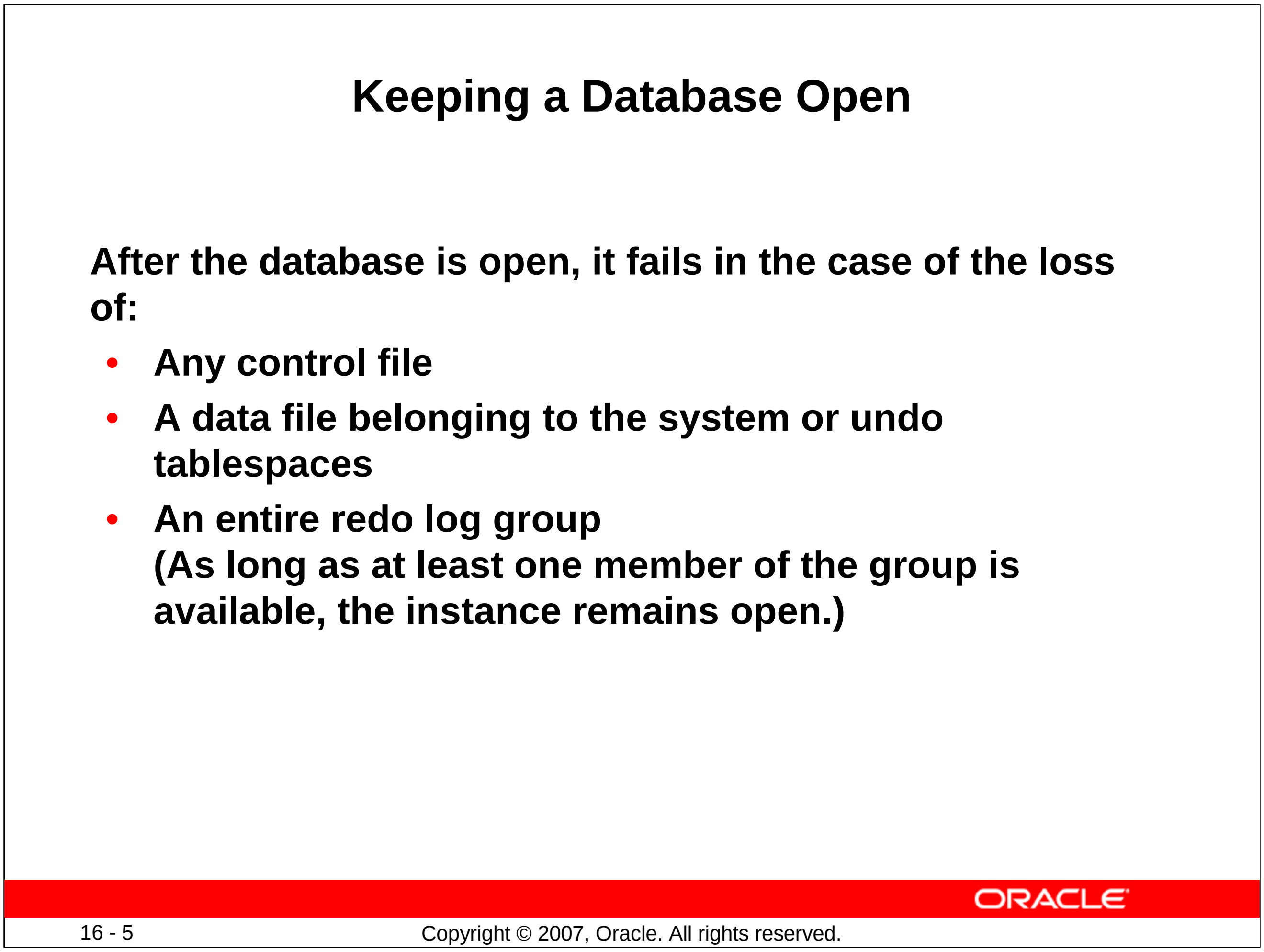
- Verifies that all data files known to the control file are present unless they have been taken offline. Offline files are not checked until the administrator tries to bring them online. The administrator may take a data file offline and open the instance if the data file does not belong to the SYSTEM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

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

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

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

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



Keeping a Database Open

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

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

16 - 5

Keeping a Database Open

After a database is open, instance failure can be caused by media failure: for example, by the loss of a control file, the loss of an entire redo log group, or the loss of a data file belonging to the `SYSTEM` or `UNDO` tablespaces.

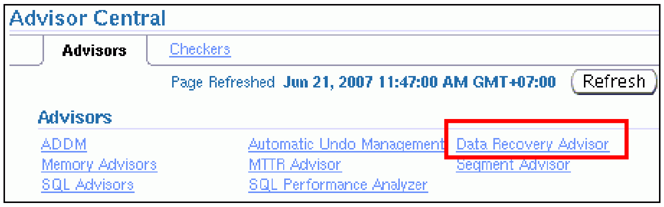
In many cases, the failed instance does not completely shut down but is unable to continue to perform work. Recovering from these types of media failure must be done with the database down. As a result, the administrator must use the `SHUTDOWN ABORT` command before beginning recovery efforts.

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

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

Data Recovery Advisor

- Fast detection, analysis, and repair of failures
- Down-time and run-time failures
- Minimizing disruptions for users
- User interfaces:
  - Enterprise Manager GUI (several paths)
  - RMAN command line



- Supported database configurations:
  - Single instance
  - Not RAC
  - Supporting failover to standby, but not analysis and repair of standby databases

Functionality of the Data Recovery Advisor

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

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

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

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

Functionality of the Data Recovery Advisor (continued)

User Interfaces

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

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

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

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

Supported Database Configurations

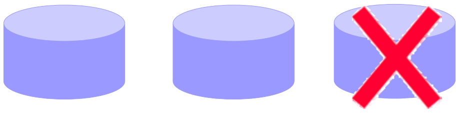
In the current release, the Data Recovery Advisor supports single-instance databases. Oracle Real Application Clusters databases are not supported.

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

Loss of a Control File

If a control file is lost or corrupted, the instance normally aborts. You must then perform the following steps:

1. **Shut down the instance (if it is still open).**
2. **Restore the missing control file by copying an existing control file.**
3. **Start the instance.**



Control files

Loss of a Control File

Recovering from the loss of a control file (if at least one control file remains) can be accomplished by performing the following steps:

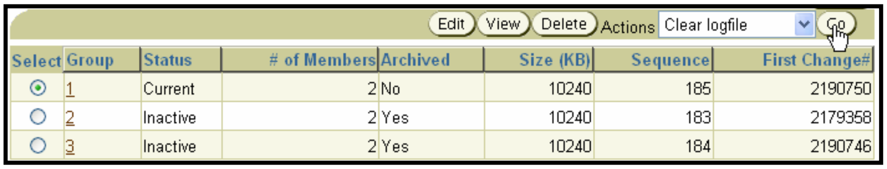
1. If the instance has not already failed, shut it down by using `SHUTDOWN ABORT`.
2. Copy one of the remaining control files to the missing file's location. If the media failure is due to the loss of a disk drive or controller, copy one of the remaining control files to some other location and update the instance's parameter file to point to the new location. Alternatively, you can delete the reference to the missing control file from the initialization parameter file. Remember that Oracle recommends having *at least* two control files at all times.
3. Start the instance.

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

Loss of a Redo Log File

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

- **Normal operation of the instance is not affected.**
- **You receive a message in the alert log notifying you that a member cannot be found.**
- **You can restore the missing log file by copying one of the remaining files from the same group.**



Select	Group	Status	# of Members	Archived	Size (KB)	Sequence	First Change#
●	1	Current	2	No	10240	185	2190750
○	2	Inactive	2	Yes	10240	183	2179358
○	3	Inactive	2	Yes	10240	184	2190746

Loss of a Redo Log File

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

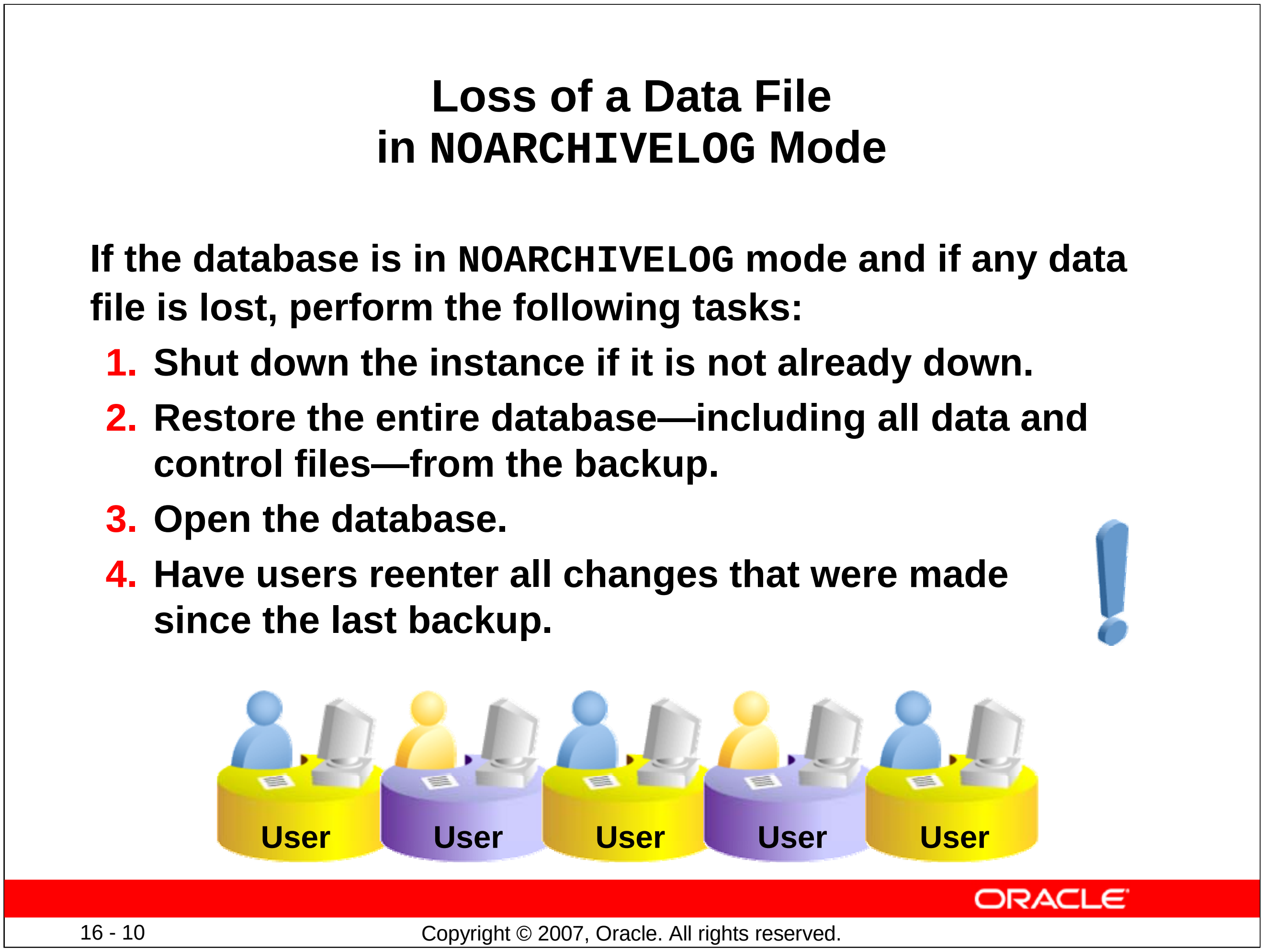
To perform this recovery:

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

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

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

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



Loss of a Data File in NOARCHIVELOG Mode

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

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

User
User
User
User
User
ORACLE

16 - 10

Loss of a Data File in NOARCHIVELOG Mode

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

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

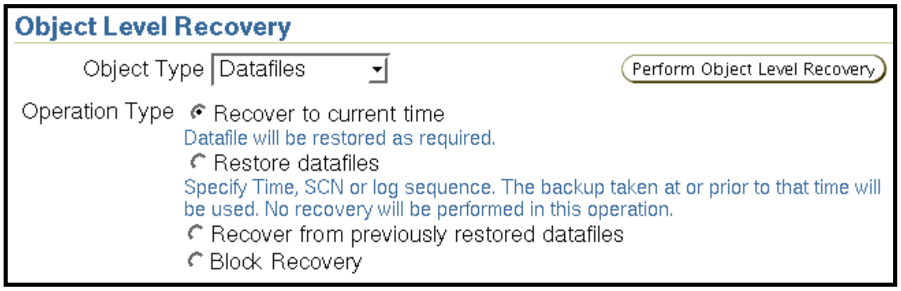
To perform this type of recovery:

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

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

Loss of a Noncritical Data File in ARCHIVELOG Mode

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



Object Level Recovery

Object Type: Datafiles

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

Users

Loss of a Noncritical Data File in ARCHIVELOG Mode

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

To restore and recover the missing data file:

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

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

Loss of a System-Critical Data File in `ARCHIVELOG` Mode

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

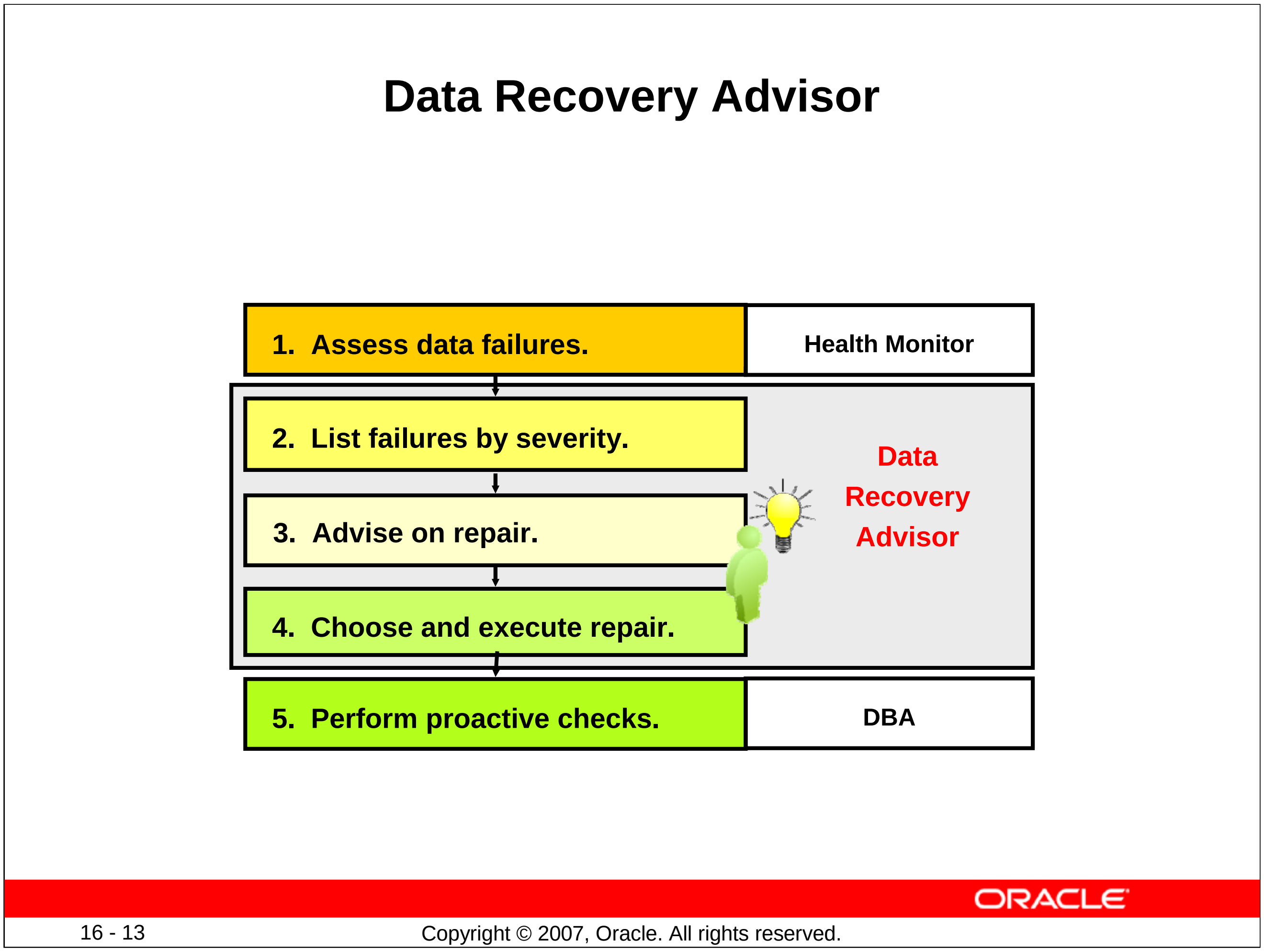
1. **The instance may or may not shut down automatically. If it does not, use `SHUTDOWN ABORT` to bring the instance down.**
2. **Mount the database.**
3. **Restore and recover the missing data file.**
4. **Open the database.**

Loss of a System-Critical Data File in `ARCHIVELOG` Mode

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

To perform this recovery:

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



Data Recovery Advisor

1. Assess data failures. | Health Monitor

2. List failures by severity.

3. Advise on repair.

4. Choose and execute repair.

Data Recovery Advisor

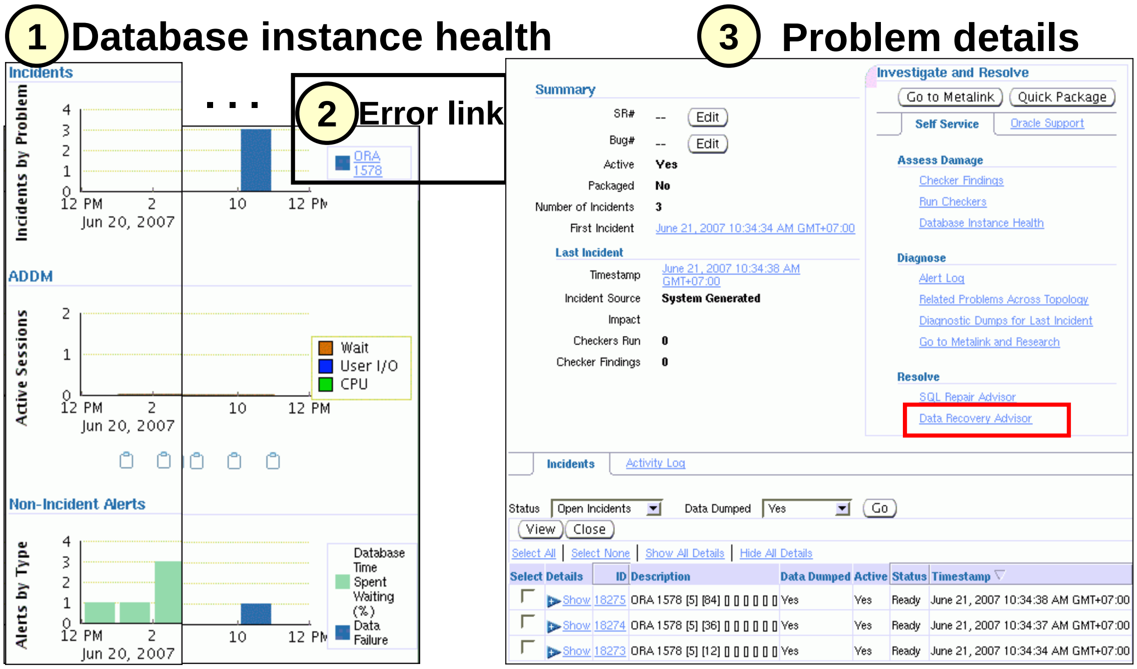
5. Perform proactive checks. | DBA

Data Recovery Advisor

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

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

Assessing Data Failures



1 Database instance health

2 Error link

3 Problem details

Summary

SR#	--	Edit
Bug#	--	Edit
Active	Yes	
Packaged	No	
Number of Incidents	3	
First Incident	June 21, 2007 10:34:34 AM GMT+07:00	

Last Incident

Timestamp	June 21, 2007 10:34:38 AM GMT+07:00
Incident Source	System Generated
Impact	
Checkers Run	0
Checker Findings	0

Investigate and Resolve

Go to Metalink | Quick Package

Self Service | Oracle Support

Assess Damage

- Checker Findings
- Run Checkers
- Database Instance Health

Diagnose

- Alert Log
- Related Problems Across Topology
- Diagnostic Dumps for Last Incident
- Go to Metalink and Research

Resolve

- SQL Repair Advisor
- Data Recovery Advisor

Incidents | Activity Log

Status: Open Incidents | Data Dumped: Yes | Go

View | Close

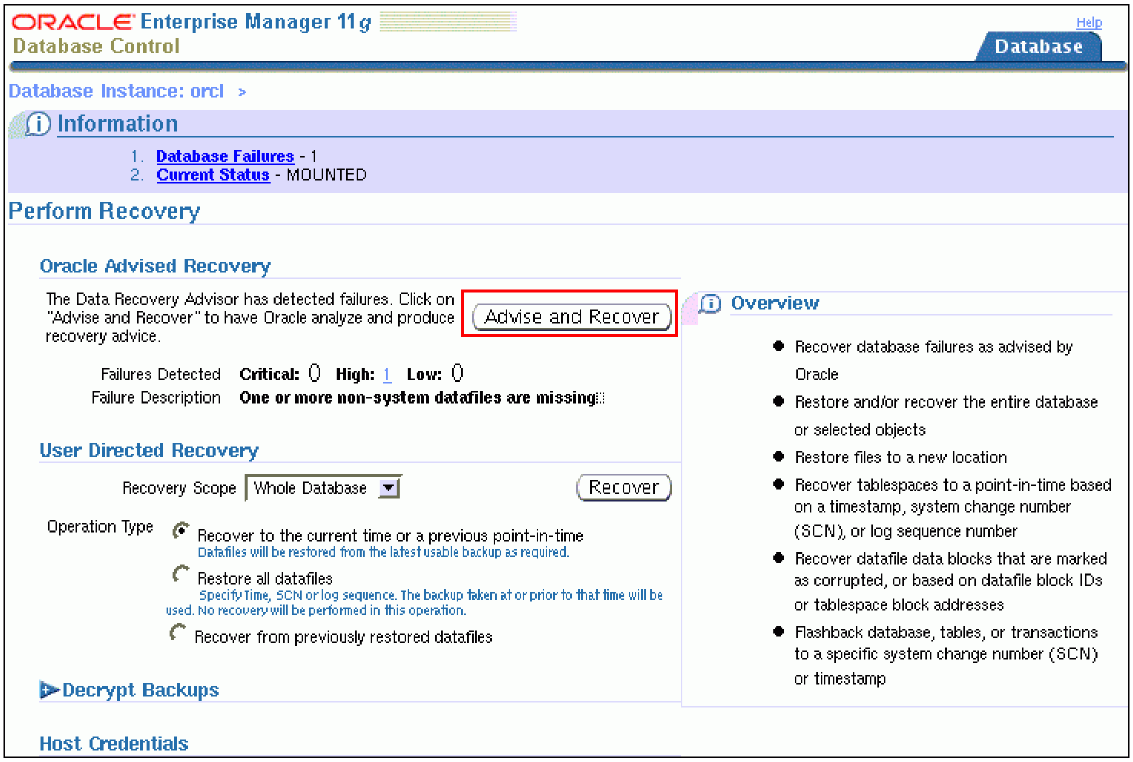
Select All | Select None | Show All Details | Hide All Details

Select	Details	ID	Description	Data Dumped	Active	Status	Timestamp ▽
□	▶ Show	18275	ORA 1578 [5] [84] ▯ ▯ ▯ ▯ ▯ ▯	Yes	Yes	Ready	June 21, 2007 10:34:38 AM GMT+07:00
□	▶ Show	18274	ORA 1578 [5] [36] ▯ ▯ ▯ ▯ ▯ ▯	Yes	Yes	Ready	June 21, 2007 10:34:37 AM GMT+07:00
□	▶ Show	18273	ORA 1578 [5] [12] ▯ ▯ ▯ ▯ ▯ ▯	Yes	Yes	Ready	June 21, 2007 10:34:34 AM GMT+07:00

Assessing Data Failures

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

Data Failures



ORACLE Enterprise Manager 11*g*
Database Control

Help

Database

Database Instance: orcl >

Information

1. Database Failures - 1
2. Current Status - MOUNTED

Perform Recovery

Oracle Advised Recovery

The Data Recovery Advisor has detected failures. Click on "Advise and Recover" to have Oracle analyze and produce recovery advice.

Advise and Recover

Failures Detected **Critical:** 0 **High:** 1 **Low:** 0

Failure Description **One or more non-system datafiles are missing**

User Directed Recovery

Recovery Scope: Whole Database ▼

Recover

Operation Type

● Recover to the current time or a previous point-in-time
Datafiles will be restored from the latest usable backup as required.

○ Restore all datafiles
Specify Time, SCN or log sequence. The backup taken at or prior to that time will be used. No recovery will be performed in this operation.

○ Recover from previously restored datafiles

▶ Decrypt Backups

Host Credentials

Overview

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

Data Failures

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

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

In Enterprise Manager, select Availability > Perform Recovery or click the Perform Recovery button if you find your database in a “down” or “mounted” state.

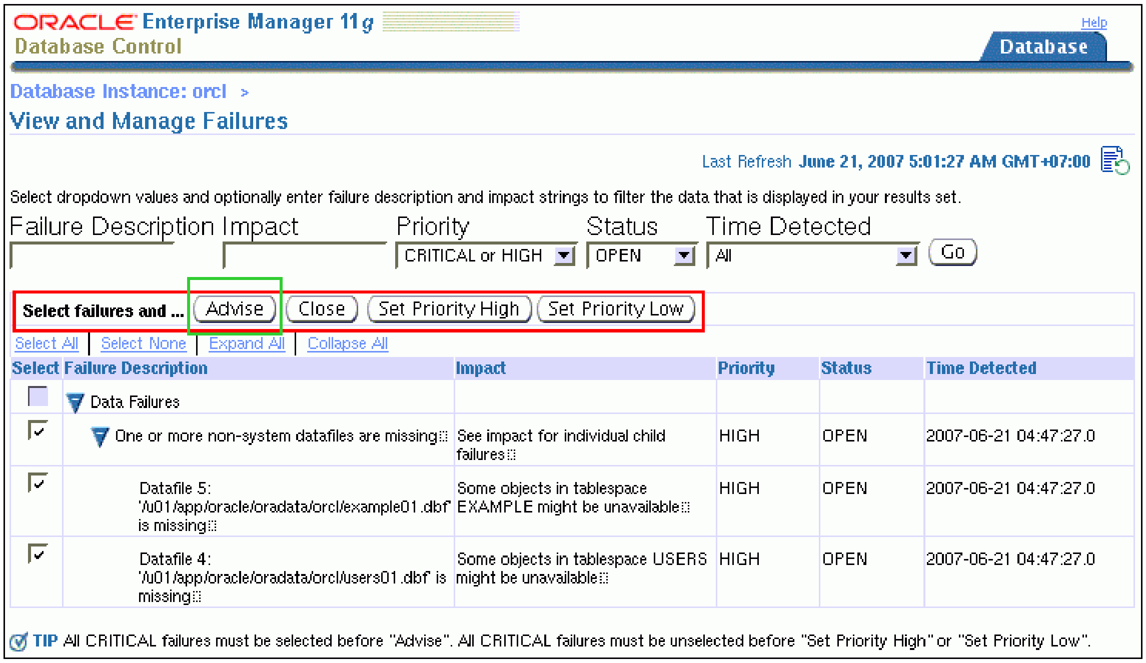
Data Failure: Examples

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

Data Failure: Examples

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

Listing Data Failures



Database Instance: orcl >

View and Manage Failures

Last Refresh June 21, 2007 5:01:27 AM GMT+07:00

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

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

Select	Failure Description	Impact	Priority	Status	Time Detected
☐	Data Failures				
☑	One or more non-system datafiles are missing	See impact for individual child failures	HIGH	OPEN	2007-06-21 04:47:27.0
☑	Datafile 5: '/u01/app/oracle/oradata/orcl/example01.dbf' is missing	Some objects in tablespace EXAMPLE might be unavailable	HIGH	OPEN	2007-06-21 04:47:27.0
☑	Datafile 4: '/u01/app/oracle/oradata/orcl/users01.dbf' is missing	Some objects in tablespace USERS might be unavailable	HIGH	OPEN	2007-06-21 04:47:27.0

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

Listing Data Failures

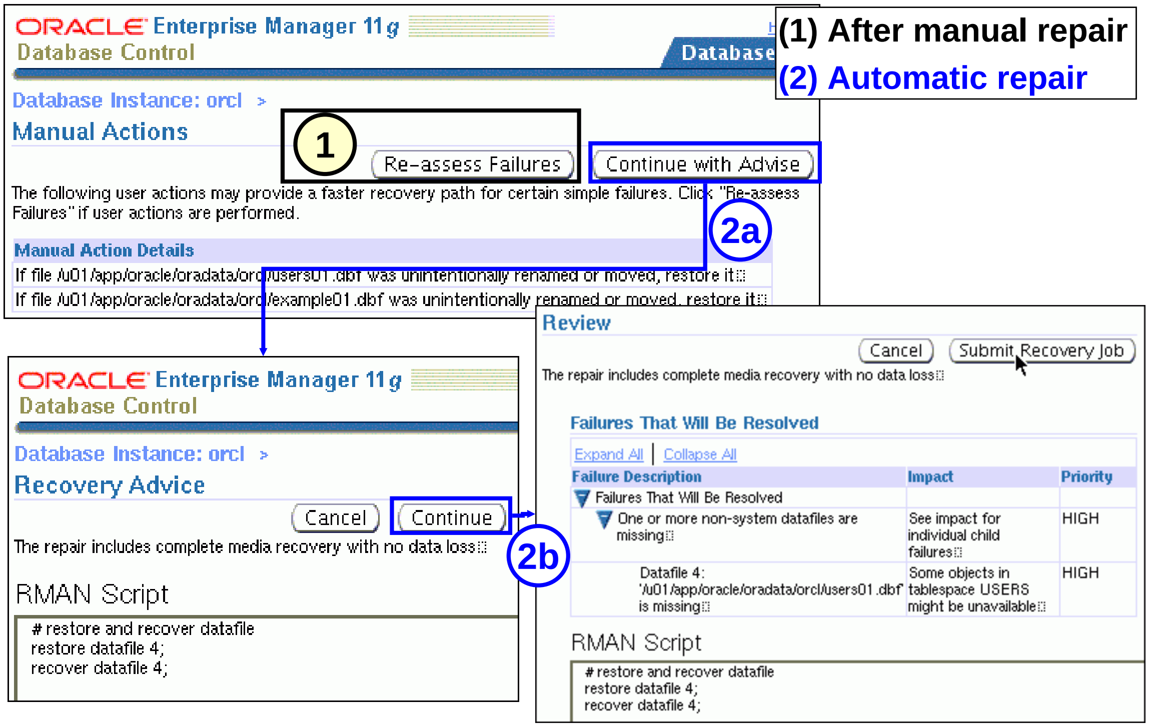
On the Perform Recovery page, click “Advise and Repair.”

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

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

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

Advising on Repair



ORACLE Enterprise Manager 11g
Database Control
Database
Database Instance: orcl >
1
Re-assess Failures
Continue with Advise
2a
ORACLE Enterprise Manager 11g
Database Control
Database Instance: orcl >
Cancel
Continue
2b
Cancel
Submit Recovery Job
Expand All | Collapse All

(1) After manual repair

(2) Automatic repair

Manual Actions

The following user actions may provide a faster recovery path for certain simple failures. Click "Re-assess Failures" if user actions are performed.

Manual Action Details
If file /u01/app/oracle/oradata/orcl/users01.dbf was unintentionally renamed or moved, restore it
If file /u01/app/oracle/oradata/orcl/example01.dbf was unintentionally renamed or moved, restore it

Recovery Advice

The repair includes complete media recovery with no data loss

RMAN Script

```
# restore and recover datafile
restore datafile 4;
recover datafile 4;
```

Review

The repair includes complete media recovery with no data loss

Failures That Will Be Resolved

Failure Description	Impact	Priority
▼ Failures That Will Be Resolved		
▼ One or more non-system datafiles are missing	See impact for individual child failures	HIGH
Datafile 4: '/u01/app/oracle/oradata/orcl/users01.dbf' is missing	Some objects in tablespace USERS might be unavailable	HIGH

RMAN Script

```
# restore and recover datafile
restore datafile 4;
recover datafile 4;
```

Advising on Repair

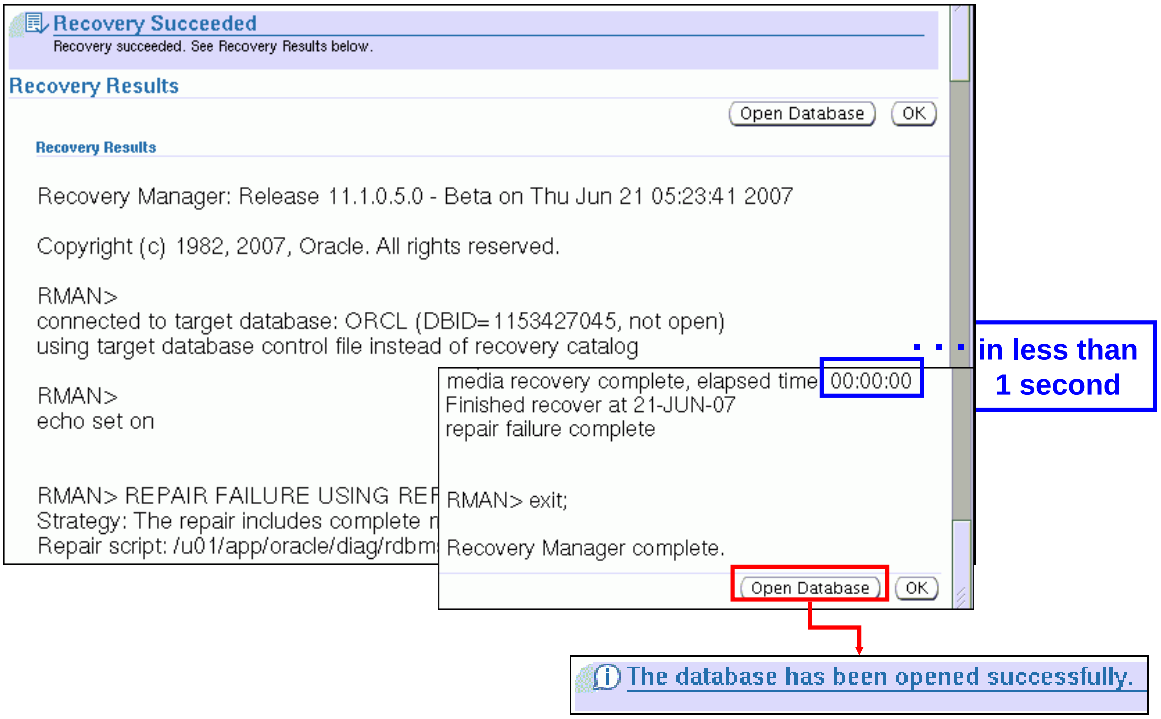
On the “View and Manage Failures” page, the Data Recovery Advisor generates a manual checklist after you click the Advise button. Two types of failures can appear.

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

You can initiate the following actions:

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

Executing Repairs



Recovery succeeded. See Recovery Results below.

```
Recovery Manager: Release 11.1.0.5.0 - Beta on Thu Jun 21 05:23:41 2007

Copyright (c) 1982, 2007, Oracle. All rights reserved.

RMAN>
connected to target database: ORCL (DBID=1153427045, not open)
using target database control file instead of recovery catalog

RMAN>
echo set on


RMAN> REPAIR FAILURE USING REF
Strategy: The repair includes complete r
Repair script: /u01/app/oracle/diag/rdbm
```

```
media recovery complete, elapsed time 00:00:00
Finished recover at 21-JUN-07
repair failure complete


RMAN> exit;

Recovery Manager complete.
```

. . . in less than 1 second

The database has been opened successfully.

Executing Repairs

The Data Recovery Advisor displays these pages. In the example, a successful repair is completed in less than one second.

Data Recovery Advisor Views

Querying dynamic data dictionary views

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

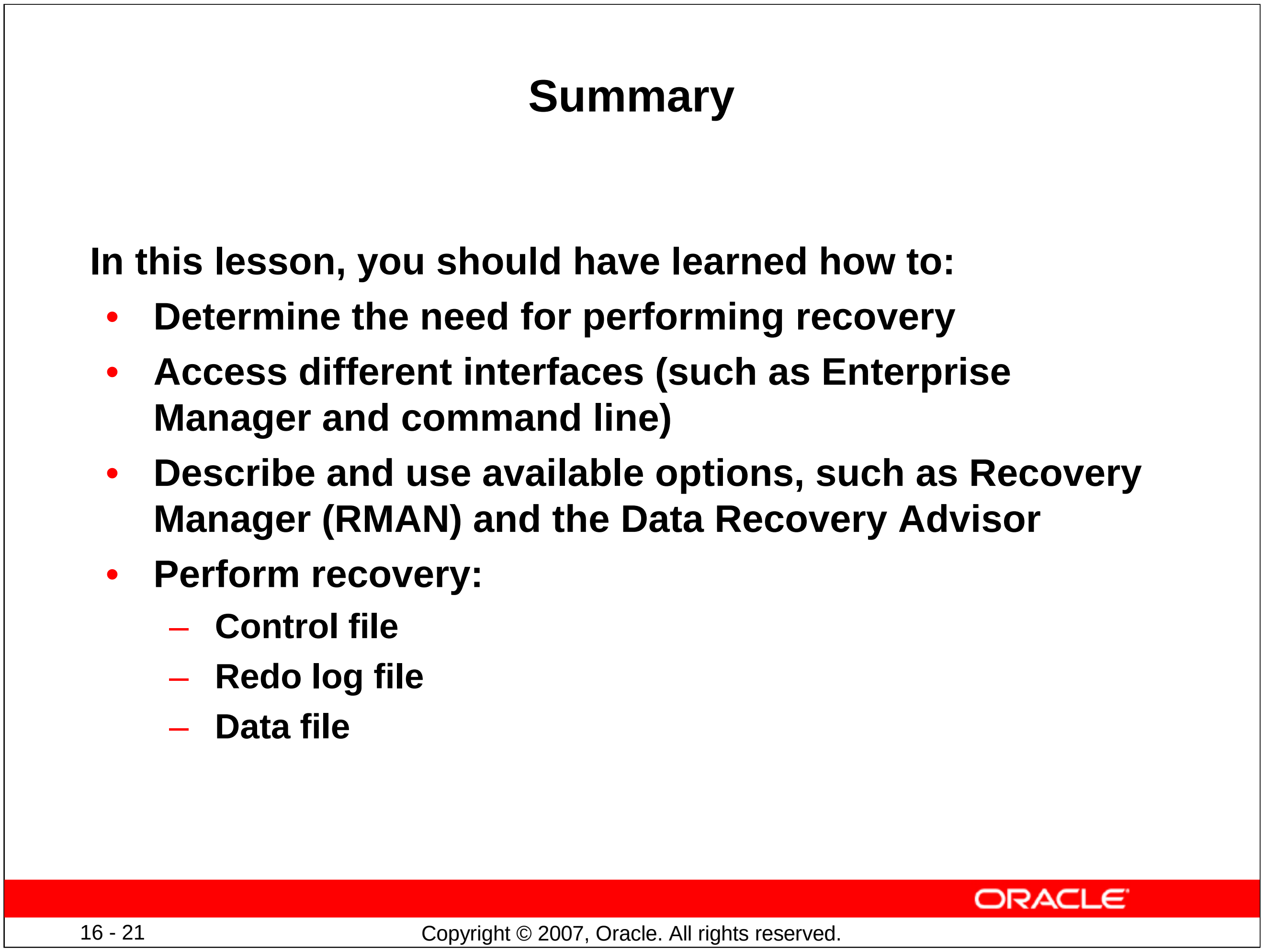
Data Recovery Advisor Views

Usage Example

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

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

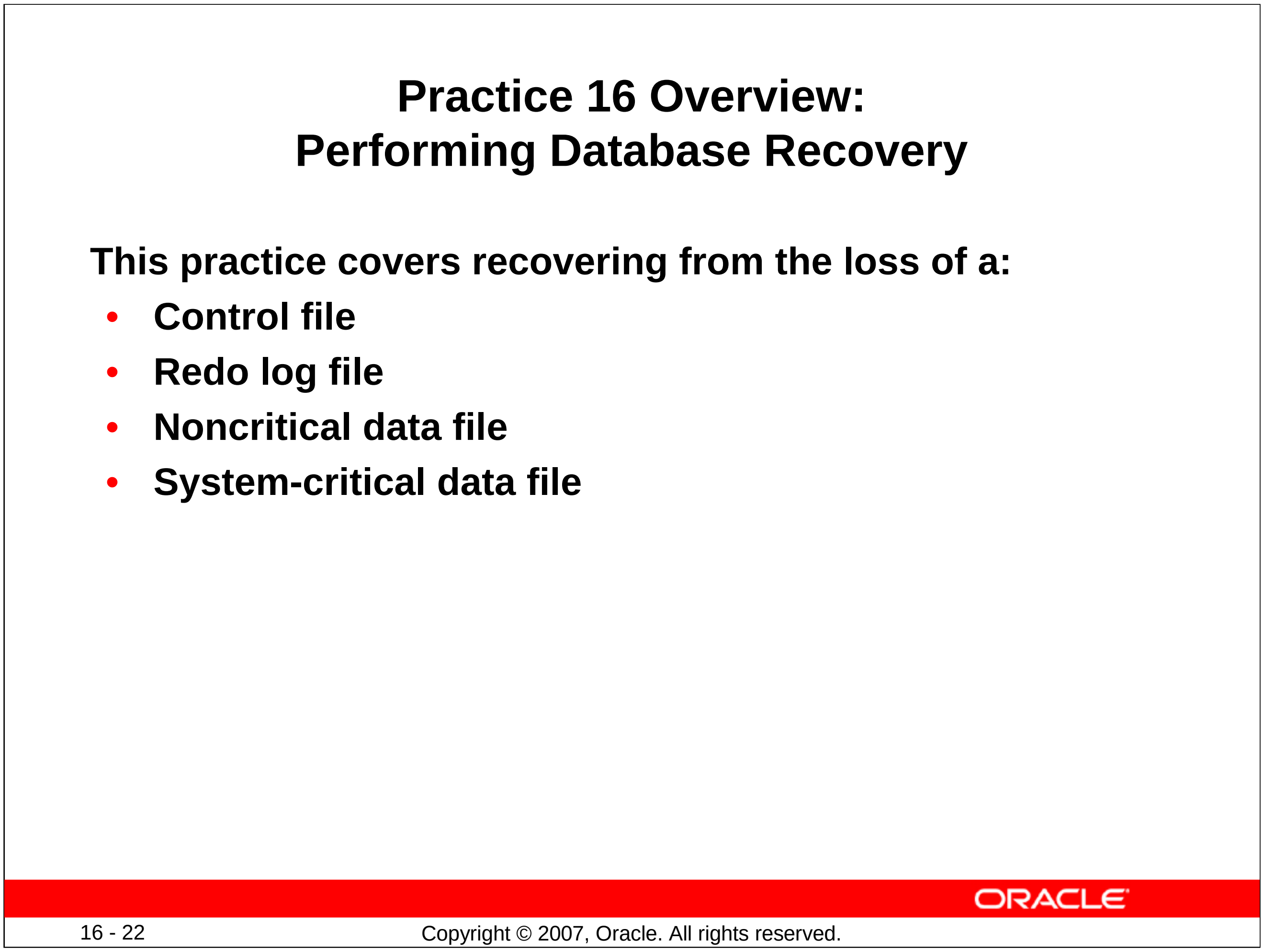
See the *Oracle Database Reference* for details about the dynamic data dictionary views used by the Data Recovery Advisor.



Summary

In this lesson, you should have learned how to:

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



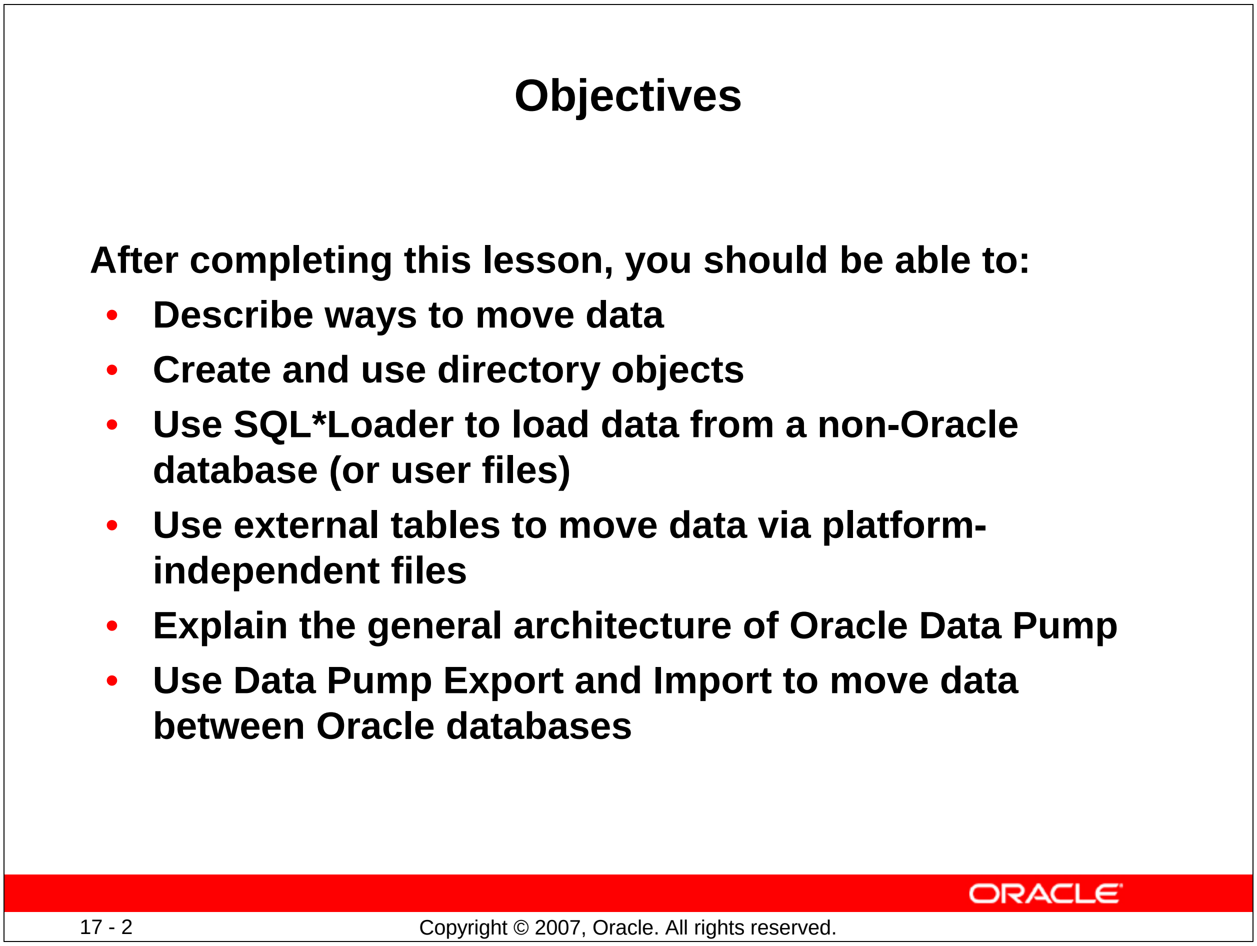
Practice 16 Overview: Performing Database Recovery

This practice covers recovering from the loss of a:

- Control file
- Redo log file
- Noncritical data file
- System-critical data file

17

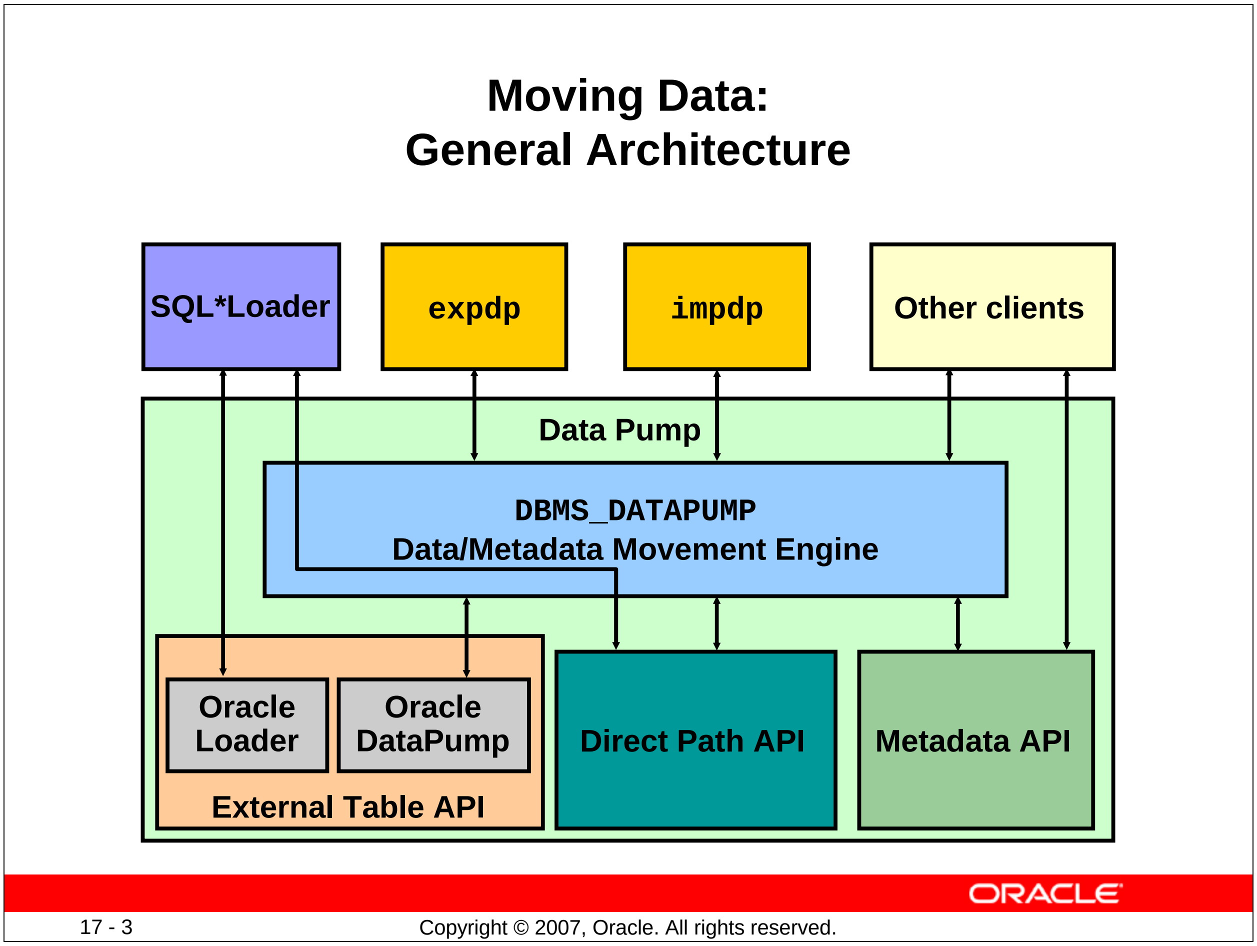
Moving Data



Objectives

After completing this lesson, you should be able to:

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



Moving Data: General Architecture

SQL*Loader

expdp

impdp

Other clients

Data Pump

DBMS_DATAPUMP
Data/Metadata Movement Engine

Oracle Loader

Oracle DataPump

External Table API

Direct Path API

Metadata API

ORACLE

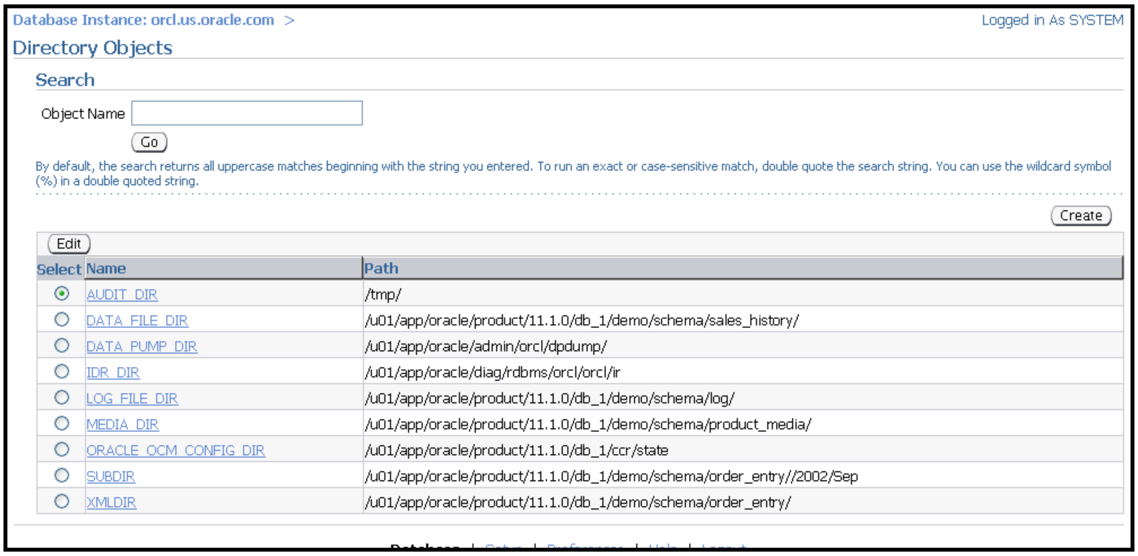
17 - 3

Moving Data: General Architecture

Major functional components:

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

Directory Objects: Overview



Database Instance: orcl.us.oracle.com >

Logged in As SYSTEM

Directory Objects

Search

Object Name

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Select	Name	Path
●	AUDIT_DIR	/tmp/
○	DATA_FILE_DIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/sales_history/
○	DATA_PUMP_DIR	/u01/app/oracle/admin/orcl/dpdump/
○	IDR_DIR	/u01/app/oracle/diag/rdbms/orcl/orcl/ir
○	LOG_FILE_DIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/log/
○	MEDIA_DIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/product_media/
○	ORACLE_OCM_CONFIG_DIR	/u01/app/oracle/product/11.1.0/db_1/ccr/state
○	SUBDIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/order_entry//2002/Sep
○	XMLDIR	/u01/app/oracle/product/11.1.0/db_1/demo/schema/order_entry/

Directory Objects: Overview

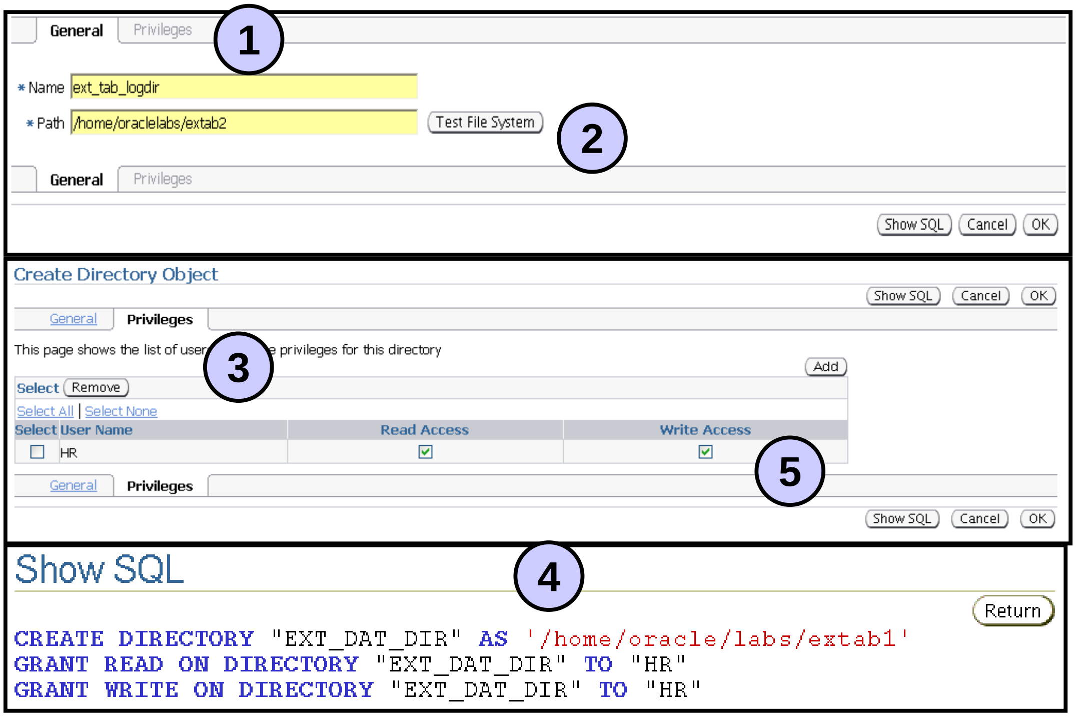
Directory objects are logical structures that represent a physical directory on the server’s file system. They contain the location of a specific operating system directory. This directory object name can be used in Enterprise Manager so that you do not need to hard-code directory path specifications.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

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

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

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

Creating Directory Objects



* Name ext_tab_logdir

* Path /home/oraclelabs/extab2

Create Directory Object

This page shows the list of user e privileges for this directory

Select	User Name	Read Access	Write Access
□	HR	☑	☑

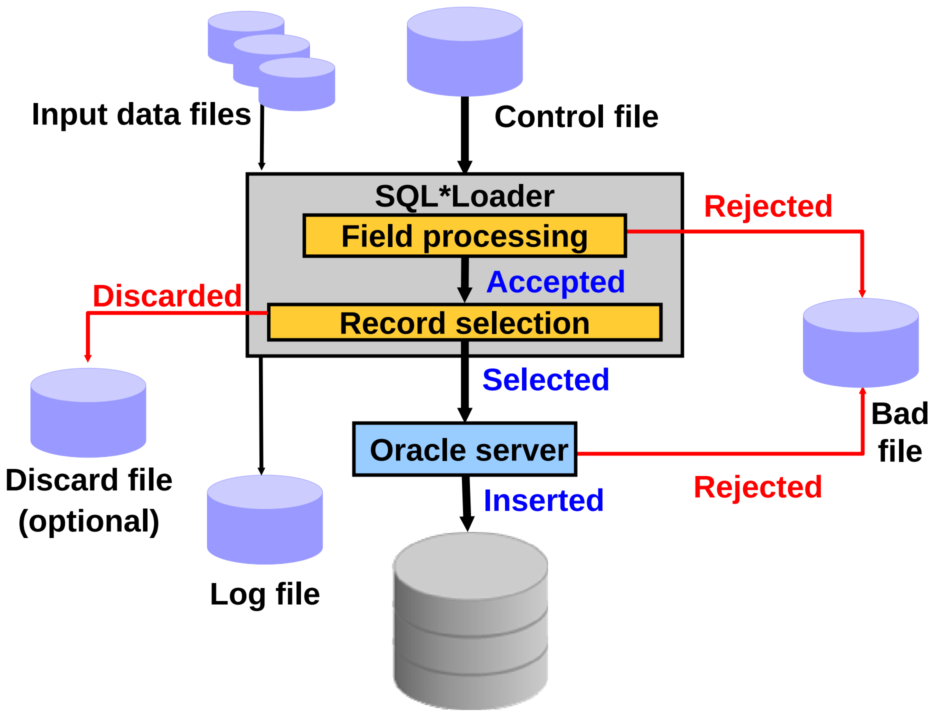
Show SQL

```
CREATE DIRECTORY "EXT_DAT_DIR" AS '/home/oracle/labs/extab1'
GRANT READ ON DIRECTORY "EXT_DAT_DIR" TO "HR"
GRANT WRITE ON DIRECTORY "EXT_DAT_DIR" TO "HR"
```

Creating Directory Objects

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

SQL*Loader: Overview



SQL*Loader: Overview

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

SQL*Loader uses the following files:

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

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

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

SQL*Loader: Overview (continued)

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

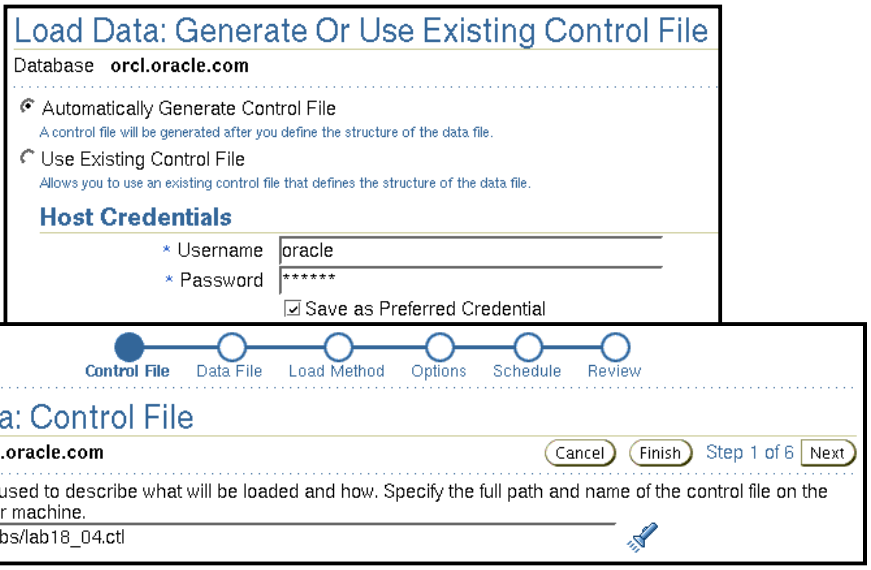
Log file: When SQL*Loader begins execution, it creates a log file. If it cannot create a log file, execution terminates. The log file contains a detailed summary of the load, including a description of any errors that occurred during the load.

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

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

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

Loading Data with SQL*Loader



Load Data: Generate Or Use Existing Control File

Database **orcl.oracle.com**

● Automatically Generate Control File

A control file will be generated after you define the structure of the data file.

○ Use Existing Control File

Allows you to use an existing control file that defines the structure of the data file.

Host Credentials

* Username oracle

* Password ******

☑ Save as Preferred Credential

Load Data: Control File

Database **orcl.oracle.com**

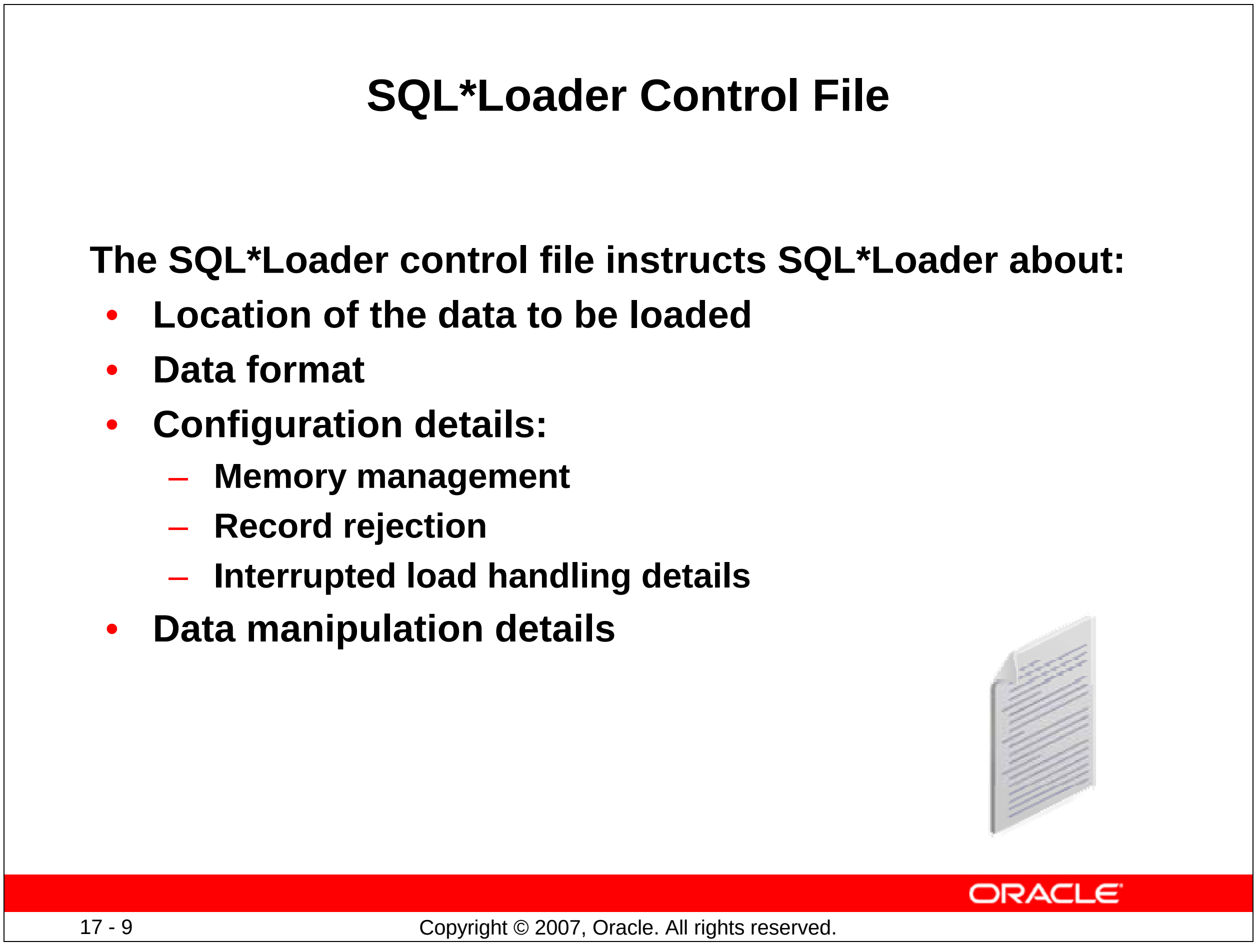
A control file is used to describe what will be loaded and how. Specify the full path and name of the control file on the database server machine.

/home/oracle/labs/lab18_04.ctl

Loading Data with SQL*Loader

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

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



SQL*Loader Control File

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

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

SQL*Loader Control File

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

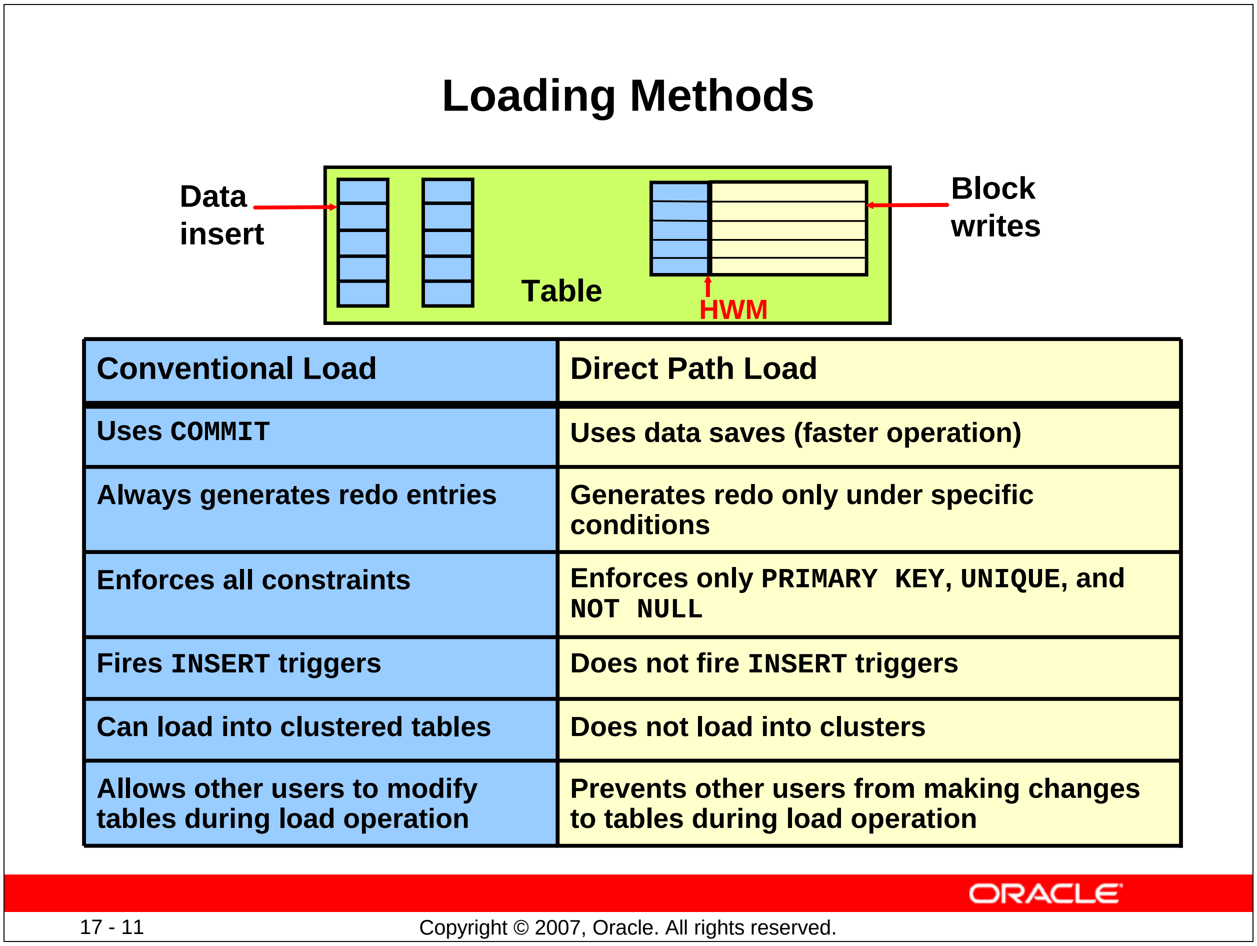
- Where SQL*Loader finds the data to load
- How SQL*Loader expects that data to be formatted
- How SQL*Loader is being configured (including memory management, selection and rejection criteria, interrupted load handling, and so on) as it loads the data
- How SQL*Loader manipulates the data being loaded

SQL*Loader Control File (continued)

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

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

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



Loading Methods

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

Comparing Direct and Conventional Path Loads

Method of Saving Data

Conventional path loads use SQL processing and a database `COMMIT` operation for saving data. The insertion of an array of records is followed by a `COMMIT` operation. Each data load may involve several transactions.

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

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

Note: Direct path and parallel direct path loads are so similar (regarding DML activities) that they are not separated in this comparison.

Comparing Direct and Conventional Path Loads (continued)

Logging Changes

Conventional path loading generates redo entries that are similar to any DML statement. When using a direct path load, redo entries are not generated if:

- The database is in `NOARCHIVELOG` mode
- The database is in `ARCHIVELOG` mode but logging is disabled
  (Logging can be disabled by setting the `NOLOGGING` attribute for the table or by using the `UNRECOVERABLE` clause in the control file.)

Enforcing Constraints

During a conventional path load, all enabled constraints are enforced in the same way that they are during any DML operation.

During direct path loads, the constraints are handled as follows:

- `NOT NULL` constraints are checked when arrays are built.
- `FOREIGN KEY` and `CHECK` constraints are disabled, and they can be enabled at the end of the load by using the appropriate commands in the control file. The `FOREIGN KEY` constraints are disabled because they reference other rows or tables, and the `CHECK` constraints are disabled because they may use SQL functions. If only a small number of rows are to be inserted into a large table, use conventional loads.
- `PRIMARY KEY` and `UNIQUE` constraints are checked during and at the end of the load, and they can be disabled if they are violated.

Firing the `INSERT` Triggers

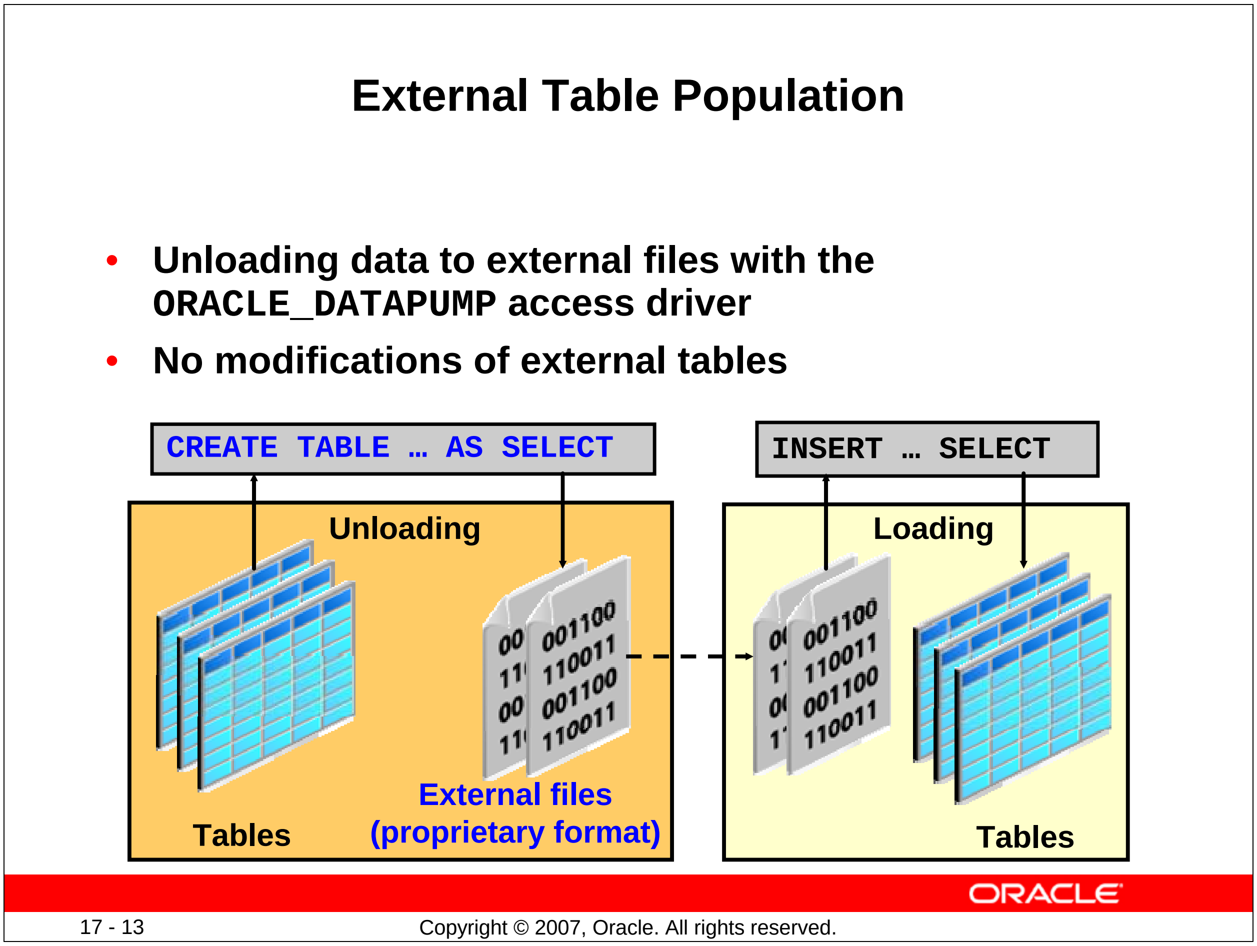
The `WHILE INSERT` triggers are fired during conventional loads; they are disabled before a direct path load and reenabled at the end of the load. They may remain disabled if a referenced object is not accessible at the end of the run. Consider using conventional path loads to load data into tables with the `INSERT` triggers.

Loading into Clustered Tables

Direct loads cannot be used to load rows into clustered tables. Clustered tables can be loaded with conventional path loads only.

Locking

While a direct path load is in progress, other transactions cannot make changes to the tables that are being loaded. The only exception to this rule is when several parallel direct load sessions are used concurrently.



External Table Population

- Unloading data to external files with the `ORACLE_DATAPUMP` access driver
- No modifications of external tables

`CREATE TABLE … AS SELECT`

`INSERT … SELECT`

Unloading

Tables

External files (proprietary format)

Loading

Tables

External Table Population

An external table is composed of proprietary format (that is, Direct Path API) flat files that are operating system independent. As data is extracted from the Oracle database and “unloaded” into files, it is transparently converted from its Oracle internal representation into an equivalent Oracle native external representation (that is, DPAPI).

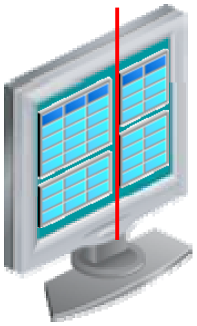
You can use the `CREATE TABLE AS SELECT` command to populate an external table. After an external table has been created and populated, no rows may be added, updated, or deleted from the external table. Any attempt to modify the data in the external table fails. An external table may not have indexes.

The Data Pump access driver enables the unloading and loading operations for external tables.

In Oracle Database 11*g*, you can compress and encrypt data before it is written to the dump file set.

Using External Tables

- Data can be used directly from the external file or loaded into another database.
- Resulting files can be read only with the `ORACLE_DATAPUMP` access driver.
- You can combine generated files from different sources for loading purposes.



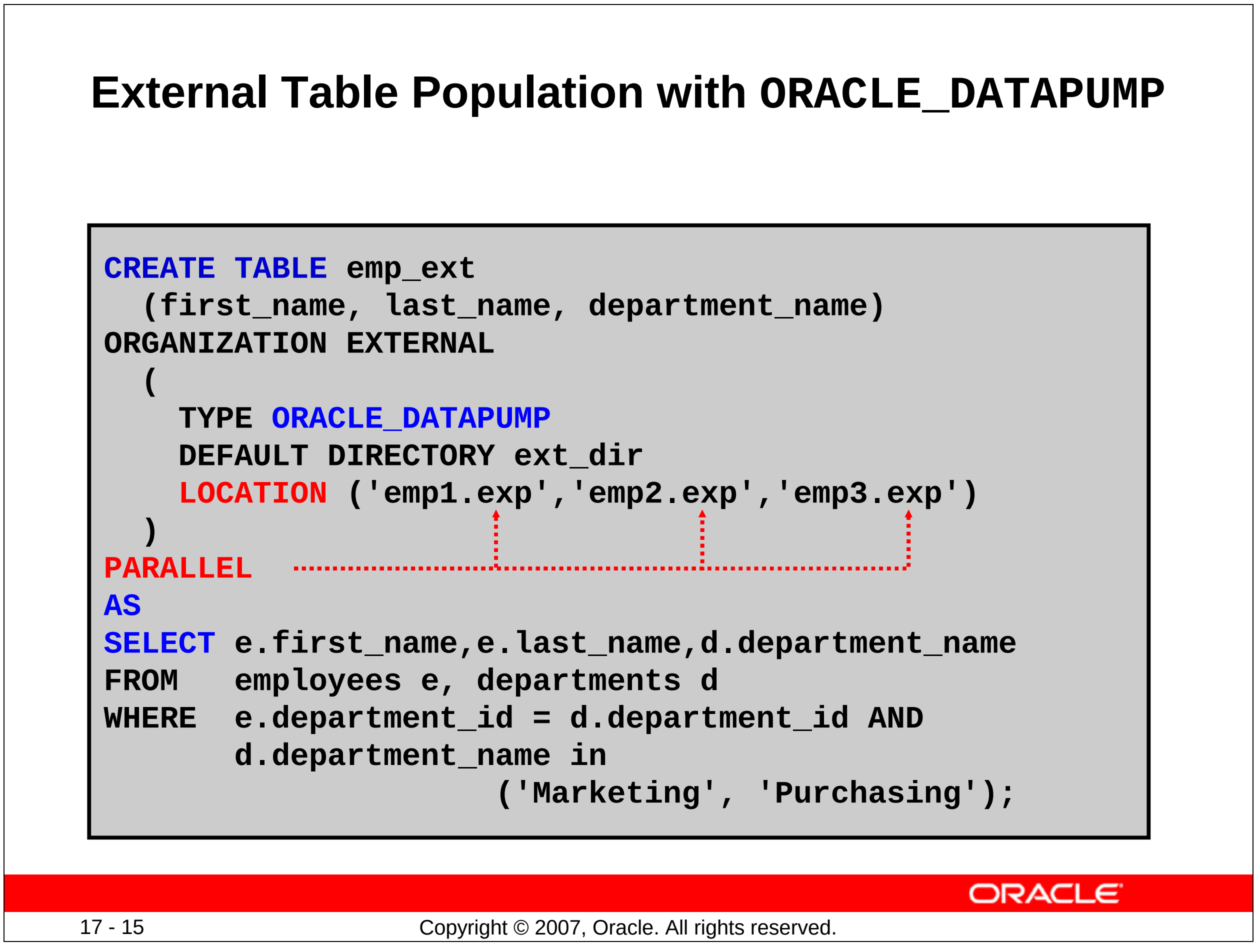
From Oracle Database

From external file

Using External Tables

The data files created for the external table can be moved and used as the data files for another external table in the same database or different database. They can be read only by the `ORACLE_DATAPUMP` access driver. You can choose to have your applications directly access external tables with the `SELECT` command, or you can choose to have data loaded first into a target database.

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



External Table Population with ORACLE_DATAPUMP

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

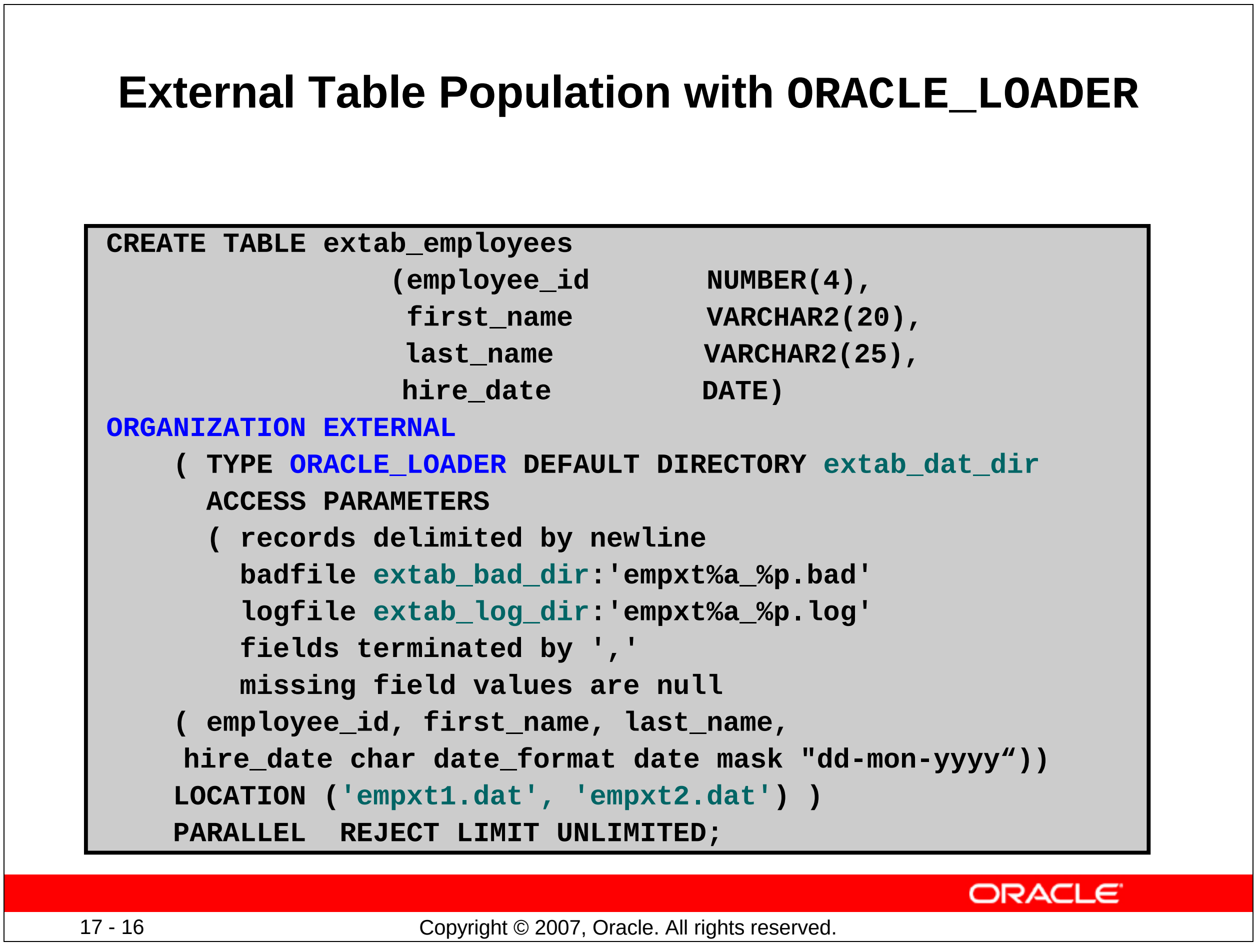
External Table Population with ORACLE_DATAPUMP

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

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

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

Note: For more information about the ORACLE_DATAPUMP access driver parameters, see the *Oracle Database Utilities* guide.



External Table Population with ORACLE_LOADER

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

17 - 16

External Table Population with ORACLE_LOADER

The ORACLE_LOADER access driver uses the SQL*Loader syntax to create external tables.

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

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

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

Oracle Data Pump: Overview

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

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

Oracle Data Pump: Overview

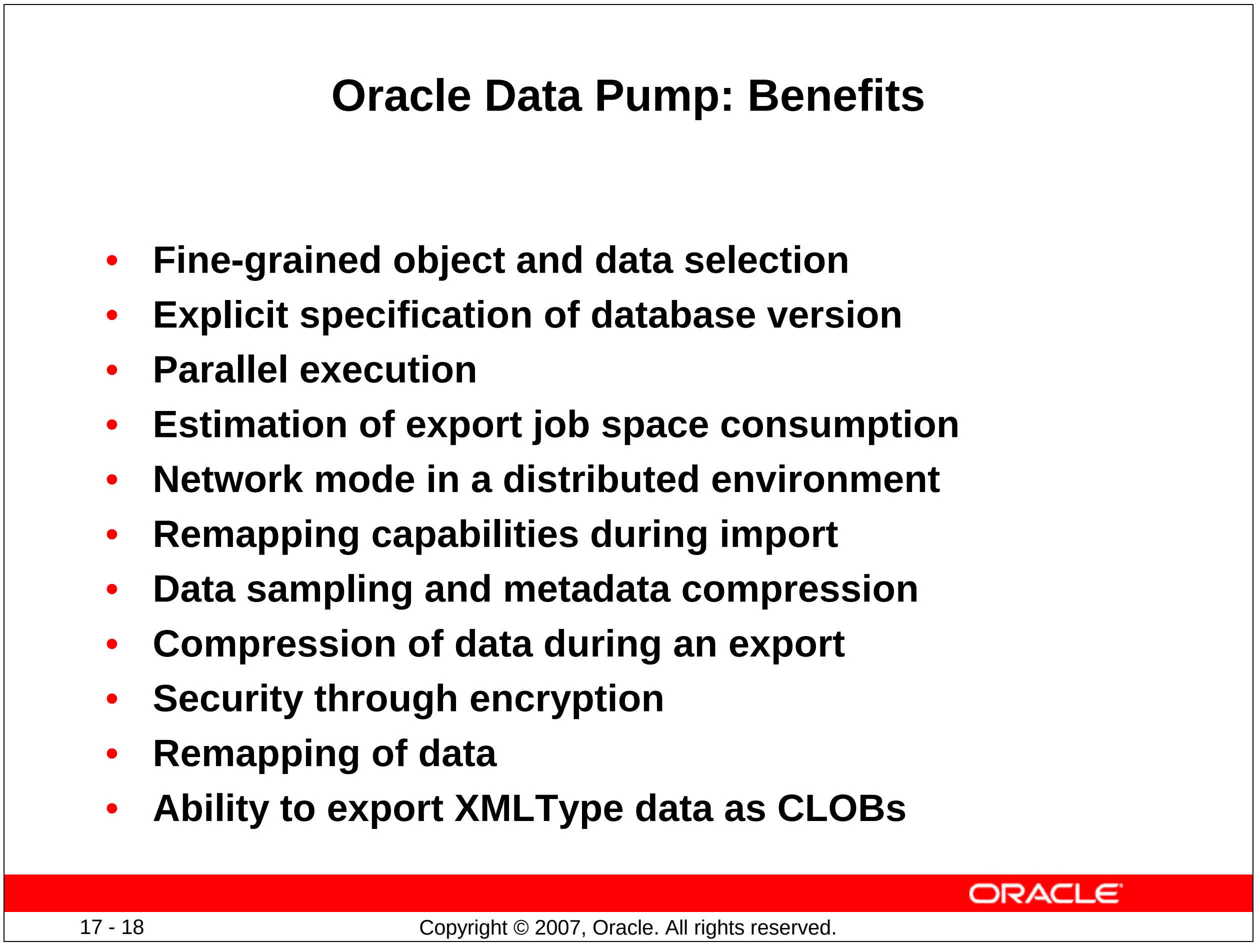
Oracle Data Pump enables very high-speed data and metadata loading and unloading of Oracle databases. The Data Pump infrastructure is callable via the `DBMS_DATAPUMP` PL/SQL package. Thus, custom data movement utilities can be built by using Data Pump.

Oracle Database 11*g* provides the following tools:

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

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

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



Oracle Data Pump: Benefits

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

Oracle Data Pump: Benefits

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

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

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

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

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

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

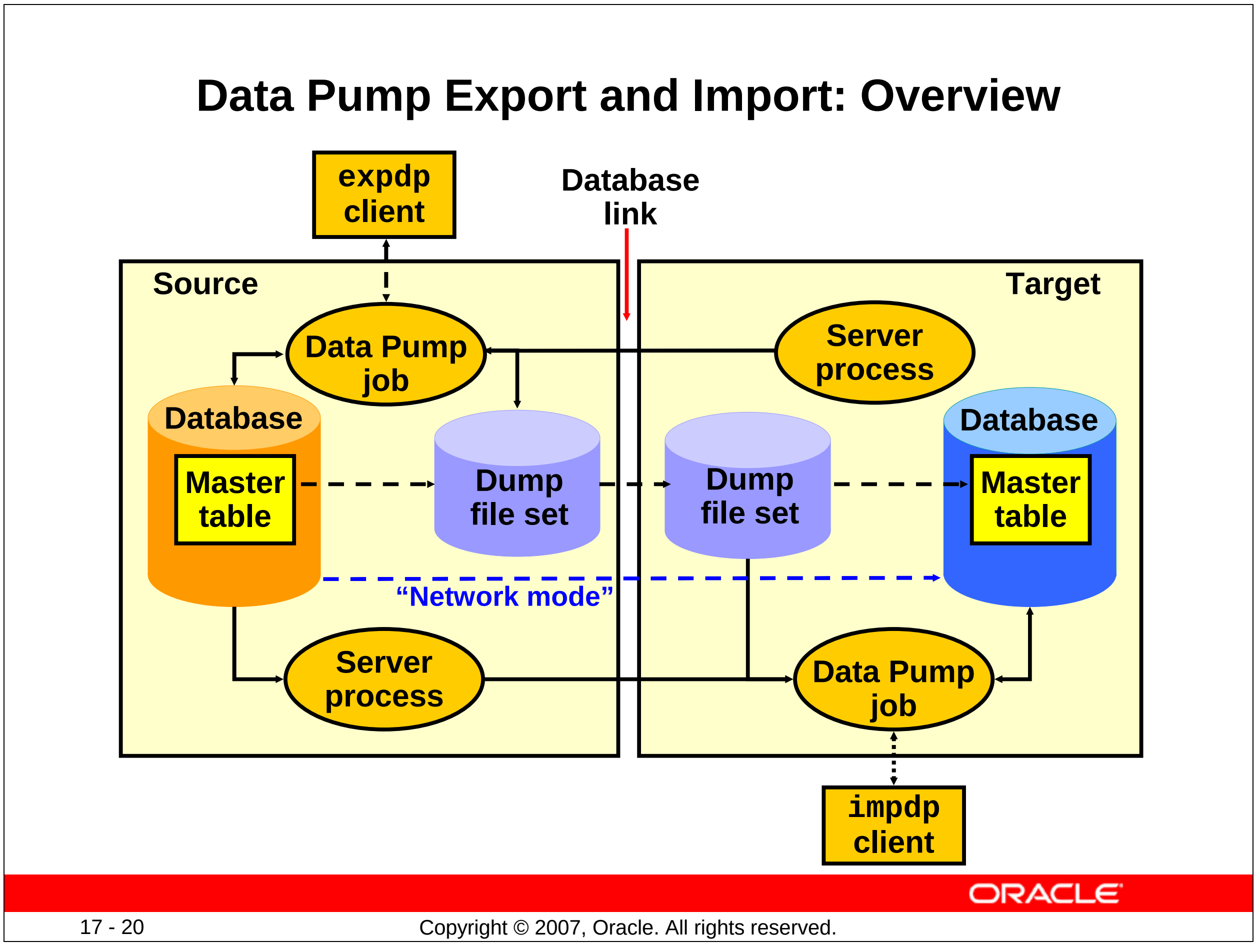
In addition you can specify a percentage of data to be sampled and unloaded from the source database when performing a Data Pump export. This can be done by specifying the `SAMPLE` parameter.

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

Data Pump Enhancements in Oracle Database 11*g*

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

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



Data Pump Export and Import: Overview

Data Pump Export and Import: Overview

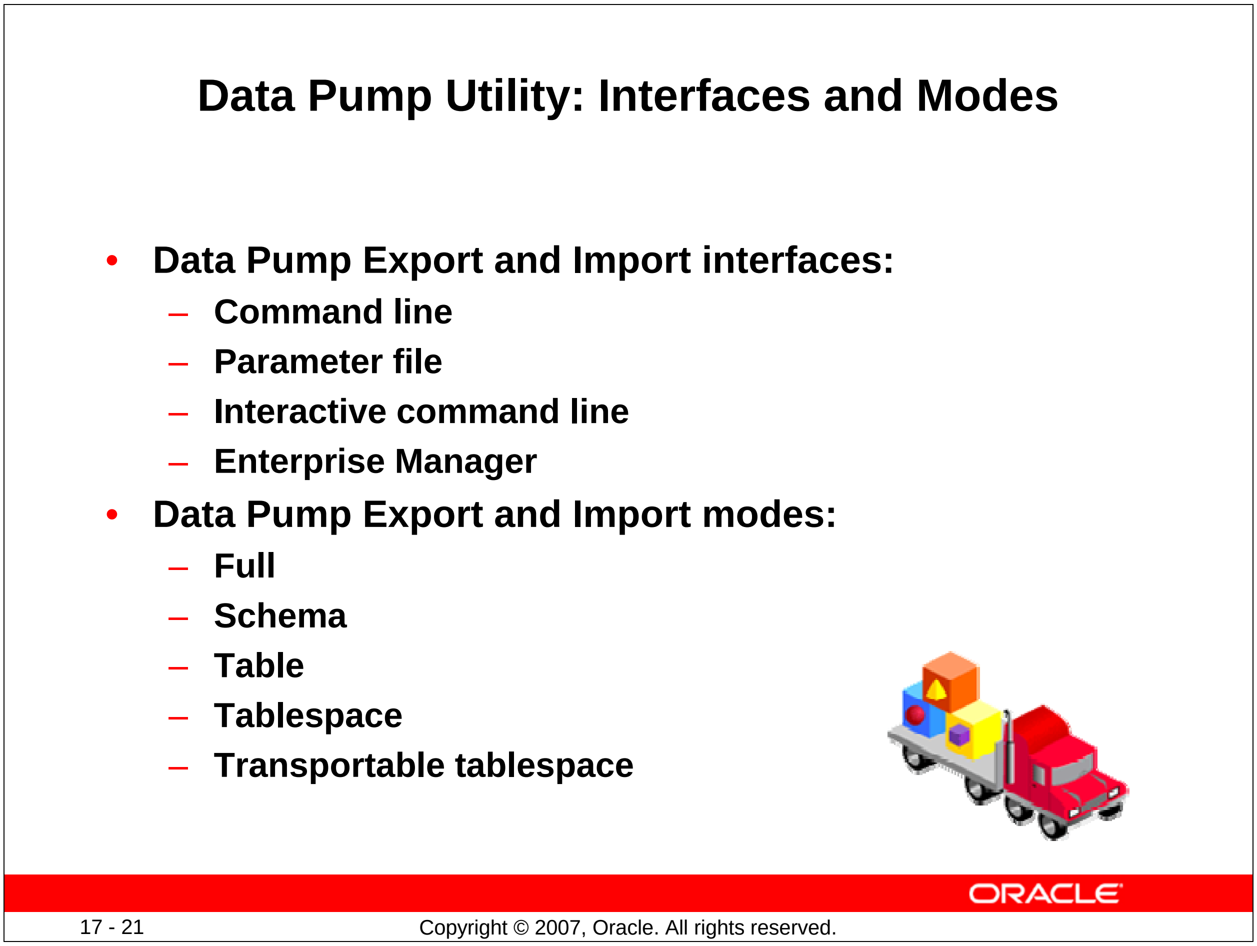
Data Pump Export is a utility for unloading data and metadata into a set of operating system files called *dump file sets*. Data Pump Import is used to load metadata and data stored in an export dump file set into a target system.

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

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

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

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



Data Pump Utility: Interfaces and Modes

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

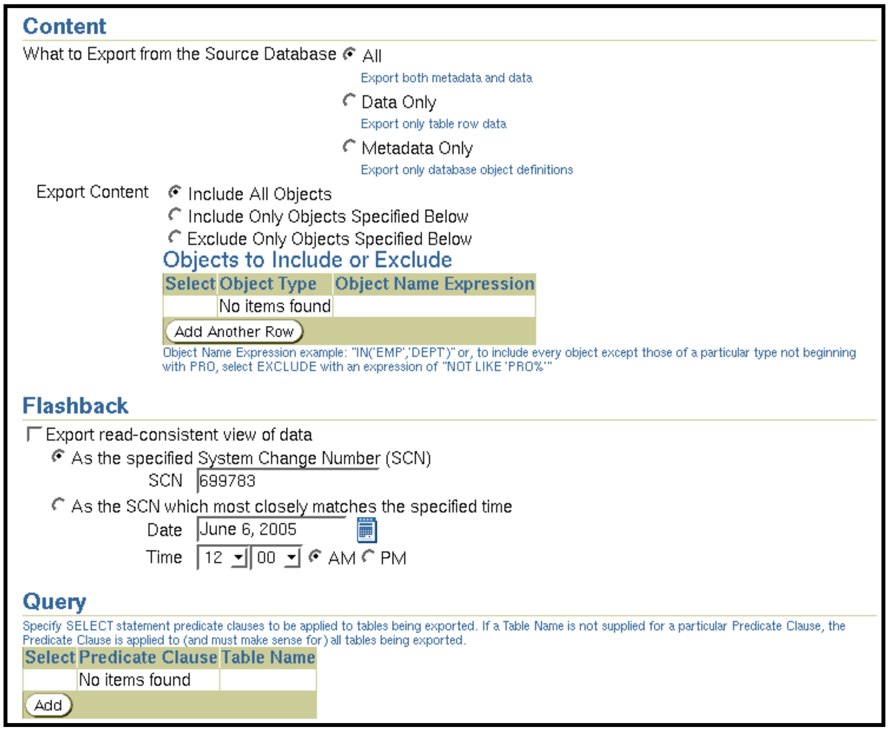
Data Pump Utility: Interfaces and Modes

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

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

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

Fine-Grained Object Selection



Content

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

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

Objects to Include or Exclude

Select	Object Type	Object Name Expression
	No items found	

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

Flashback

□ Export read-consistent view of data
● As the specified System Change Number (SCN)
SCN 699783
○ As the SCN which most closely matches the specified time
Date June 6, 2005
Time 12 00 ● AM ○ PM

Query

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

Select	Predicate Clause	Table Name
	No items found	

Fine-Grained Object Selection

The Data Pump job can include or exclude virtually any type of object.

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

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

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

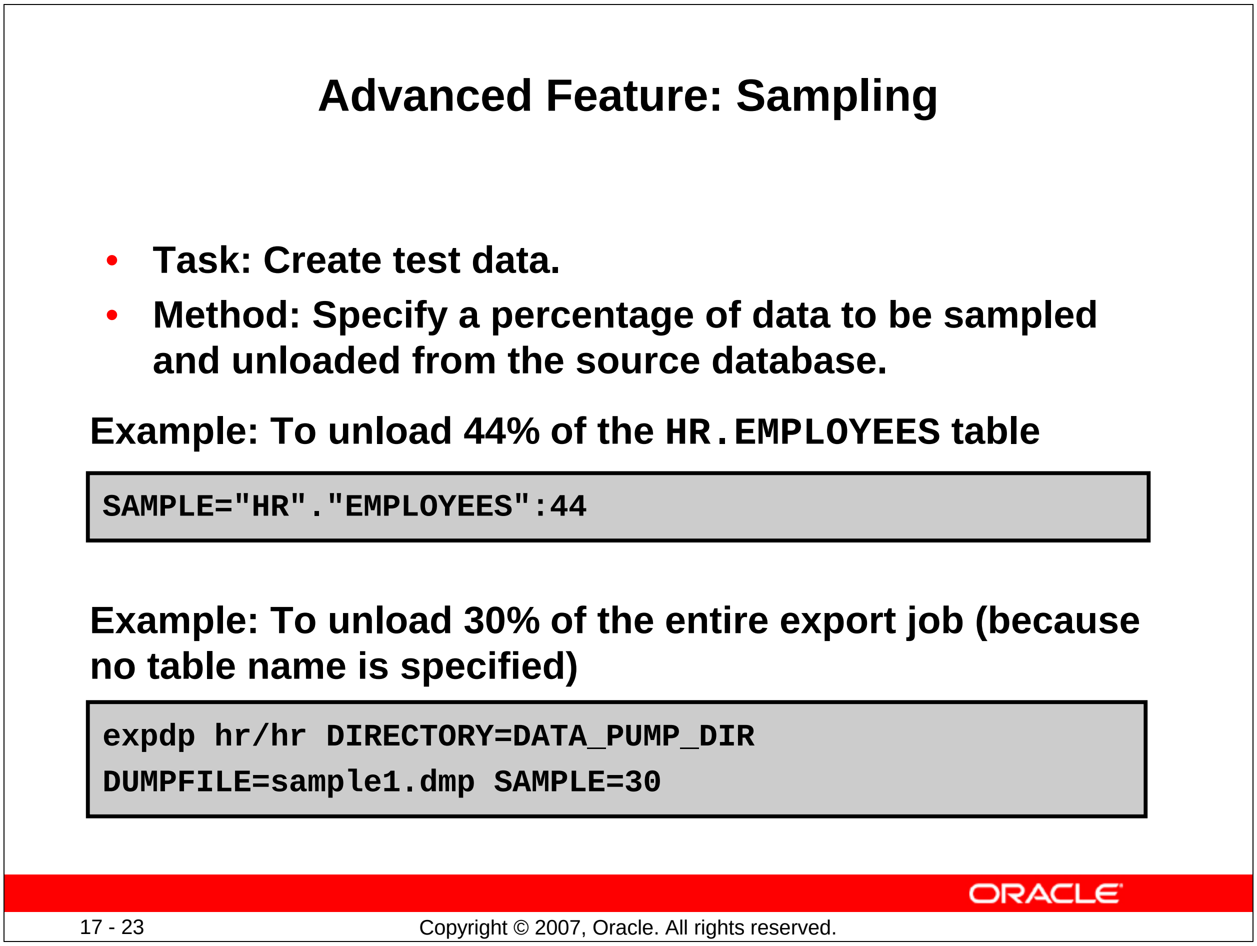
Syntax: `INCLUDE  = object_type[:"name_expr"]`

The CONTENT parameter enables you to request the current operation, only the metadata, only the data, or both metadata and data.

Syntax: `CONTENT = ALL | METADATA_ONLY | DATA_ONLY`

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

```
QUERY=hr.employees:"WHERE department_id in (10,20) and
salary < 1600 ORDER BY department_id"
```



Advanced Feature: Sampling

- **Task: Create test data.**
- **Method: Specify a percentage of data to be sampled and unloaded from the source database.**

Example: To unload 44% of the `HR.EMPLOYEES` table

```
SAMPLE="HR"."EMPLOYEES":44
```

Example: To unload 30% of the entire export job (because no table name is specified)

```
expdp hr/hr DIRECTORY=DATA_PUMP_DIR
DUMPFILE=sample1.dmp SAMPLE=30
```

Advanced Feature: Sampling

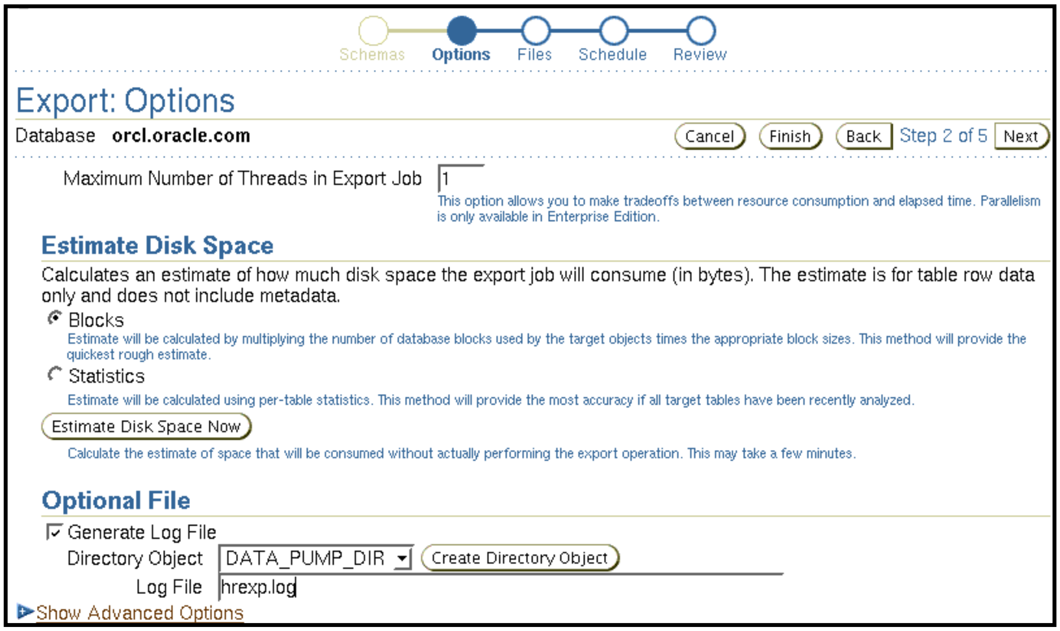
With the `SAMPLE` parameter, you can specify a percentage of data to be sampled and unloaded from the source database when performing a Data Pump export.

Syntax: `SAMPLE=`*`[[schema_name.]table_name:]sample_percent`*

The sample percentage indicates the likelihood that a block of rows will be included. The range of values for *`sample_percent`* is `.000001` to (but not including) `100`.

Note: The `SAMPLE` parameter is not valid for network exports.

Export Options: Files



Export: Options

Database **orcl.oracle.com**

Maximum Number of Threads in Export Job: 1

This option allows you to make tradeoffs between resource consumption and elapsed time. Parallelism is only available in Enterprise Edition.

Estimate Disk Space

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

● Blocks

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

○ Statistics

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

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

Optional File

☑ Generate Log File

Directory Object: DATA_PUMP_DIR

Log File: hrexp.log

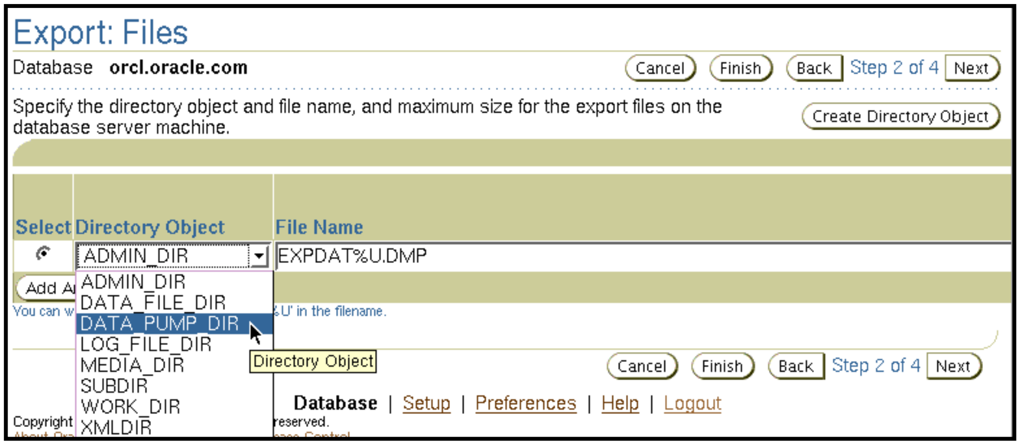
Export Options: Files

Three types of files are managed by Data Pump jobs:

- Dump files for data and metadata that is to be moved
- Log files for messages
- SQL files for the output of a `SQLFILE` operation

Because Data Pump is server based rather than client based, Data Pump files are accessed relative to Oracle directory paths. Absolute paths are not supported for security reasons.

Data Pump File Locations



Export: Files

Database orcl.oracle.com

Specify the directory object and file name, and maximum size for the export files on the database server machine.

Select	Directory Object	File Name
●	ADMIN_DIR	EXPDAT%U.DMP

You can w… %U' in the filename.

Copyright … reserved.

Order of precedence of file locations:

- **Per-file directory**
- **DIRECTORY parameter**
- **DATA_PUMP_DIR environment variable**
- **DATA_PUMP_DIR directory object**

ORACLE

Data Pump File Locations

The slide shows you the order of precedence used by Data Pump clients to locate files.

- Per-file directory objects may be specified for each dump file, log file, and SQL file. If specified, they are separated from the file name by a colon (:).
- The Data Pump Export and Import clients provide a DIRECTORY parameter, which specifies the name of a directory object. These directory objects describe the location in which the files are accessed.
- You can alternatively define an environment variable, DATA_PUMP_DIR, to specify the directory object name rather than use the DIRECTORY parameter. The Data Pump clients look for this environment variable if no explicit directory object is specified.
- A default directory object is created for every database. This directory object is named DATA_PUMP_DIR. Access to the DATA_PUMP_DIR directory is granted automatically to the EXP_FULL_DATABASE and IMP_FULL_DATABASE roles.

Data Pump File Locations (continued)

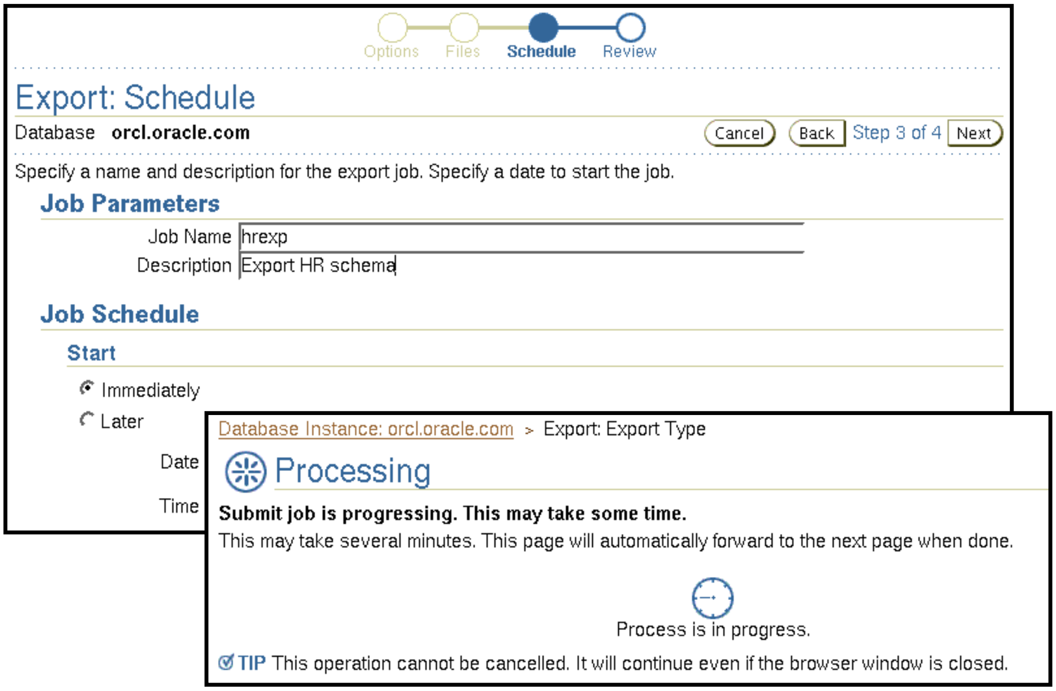
- You do not need to create a directory object manually before using Data Pump Export. A default directory object named DATA_PUMP_DIR is created for every database, whether newly created or upgraded by a script on UNIX or Windows platforms. Access to the DATA_PUMP_DIR directory is granted automatically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

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

  ```
      ORACLE_HOME/rdbms/log
  ```

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

Scheduling and Running a Job



Export: Schedule

Database orcl.oracle.com

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

Job Parameters

Job Name: hrexp

Description: Export HR schema

Job Schedule

Start

● Immediately

○ Later

Date

Time

Database Instance: orcl.oracle.com > Export: Export Type

Processing

Submit job is progressing. This may take some time.

This may take several minutes. This page will automatically forward to the next page when done.

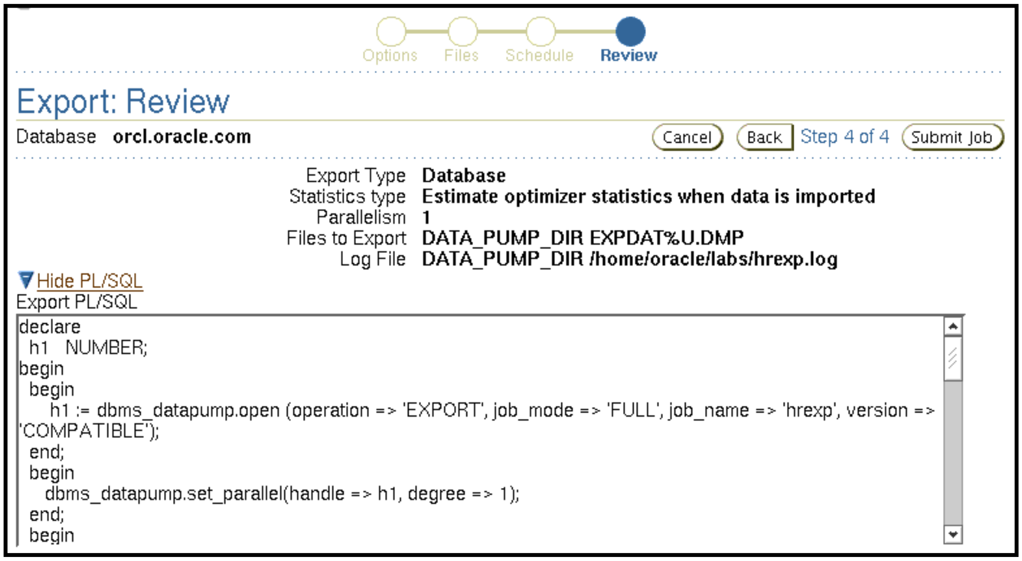
Process is in progress.

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

Scheduling and Running a Job

Data Pump jobs (created through this wizard) are scheduled as repeatable jobs by Enterprise Manager Database Control.

Data Pump File Naming and Size



Export: Review

Database orcl.oracle.com

Export Type	Database
Statistics type	Estimate optimizer statistics when data is imported
Parallelism	1
Files to Export	DATA_PUMP_DIR EXPDAT%U.DMP
Log File	DATA_PUMP_DIR /home/oracle/labs/hrexp.log

Export PL/SQL

```
declare
  h1   NUMBER;
begin
  begin
     h1 := dbms_datapump.open (operation => 'EXPORT', job_mode => 'FULL', job_name => 'hrexp', version =>
'COMPATIBLE');
  end;
  begin
     dbms_datapump.set_parallel(handle => h1, degree => 1);
  end;
  begin
```

Data Pump File Naming and Size

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

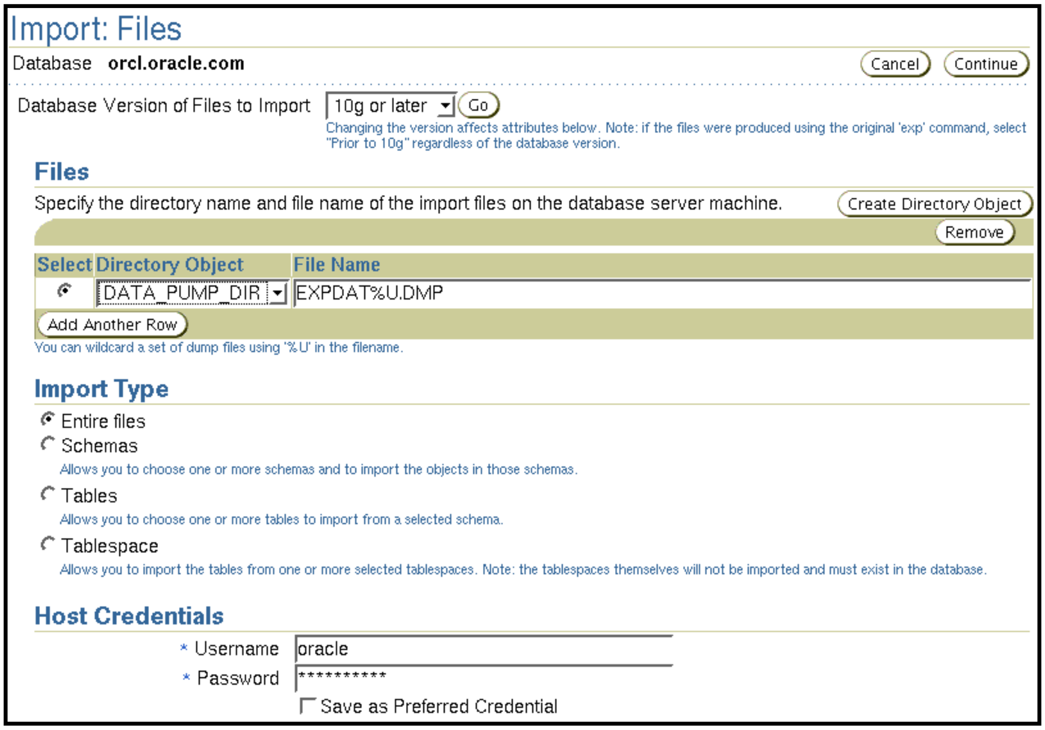
If FILESIZE is specified, each file is FILESIZE bytes in size and nonextensible. If more dump space is required and a template with %U has been supplied, a new file is automatically created with FILESIZE bytes; otherwise, the client receives a message to add a new file.

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

Preexisting files that match the resulting file names are not overwritten. Instead, they result in an error and cause the job to be aborted.

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

Data Pump Import



Import: Files

Database **orcl.oracle.com**

Database Version of Files to Import: 10g or later

Changing the version affects attributes below. Note: if the files were produced using the original 'exp' command, select "Prior to 10g" regardless of the database version.

Files

Specify the directory name and file name of the import files on the database server machine.

Select	Directory Object	File Name
●	DATA_PUMP_DIR	EXPDAT%U.DMP

You can wildcard a set of dump files using '%U' in the filename.

Import Type

● Entire files

○ Schemas

Allows you to choose one or more schemas and to import the objects in those schemas.

○ Tables

Allows you to choose one or more tables to import from a selected schema.

○ Tablespace

Allows you to import the tables from one or more selected tablespaces. Note: the tablespaces themselves will not be imported and must exist in the database.

Host Credentials

* Username: oracle

* Password: **********

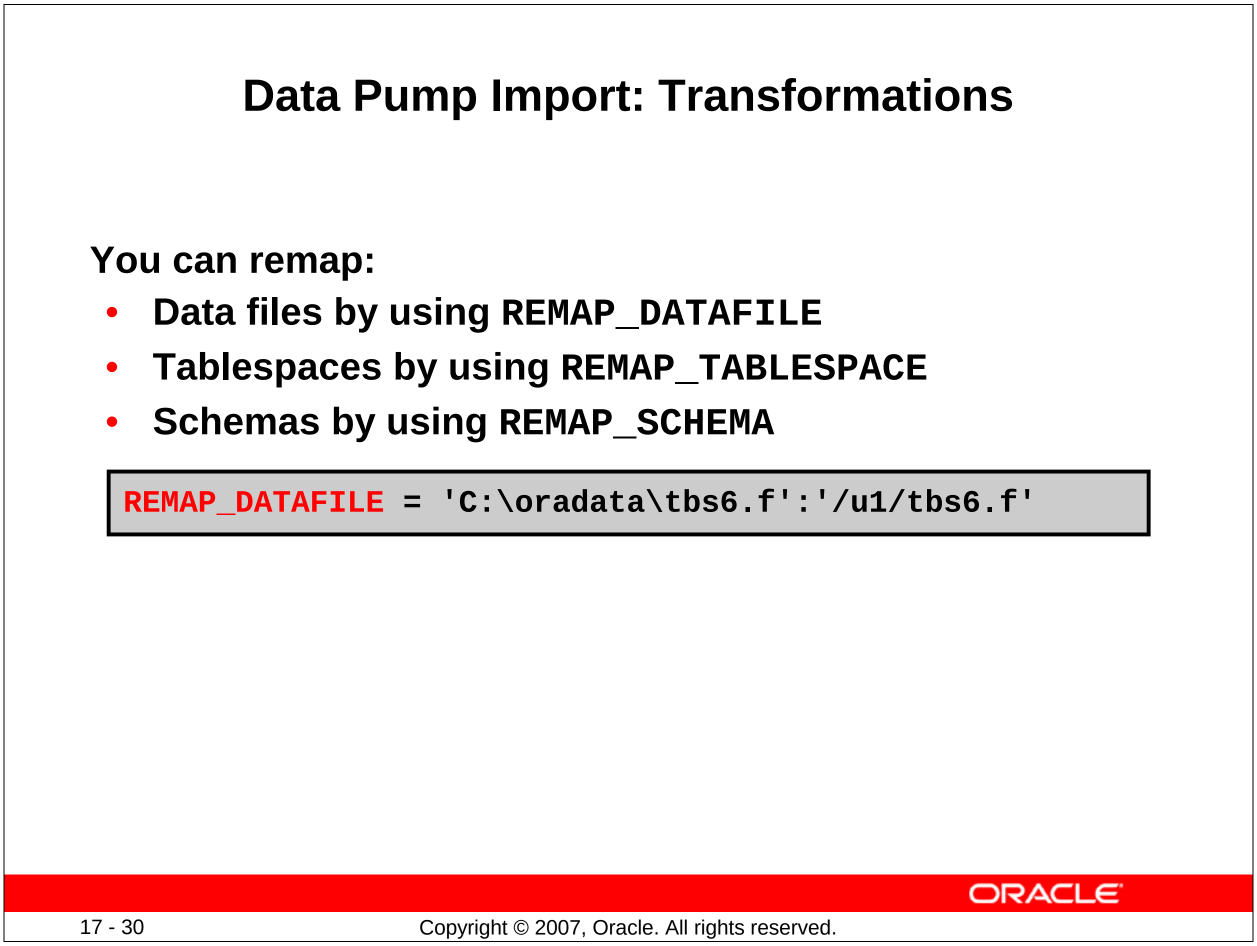
□ Save as Preferred Credential

Data Pump Import

Data Pump Import is a utility for loading an export dump file set into a target system. The dump file set comprises one or more disk files that contain table data, database object metadata, and control information. The files are written in a proprietary binary format. During an import operation, Data Pump Import uses these files to locate each database object in the dump file set.

You can interact with Data Pump Import by using a command line, a parameter file, or an interactive-command mode:

- You can use the `impdp` command and specify parameters directly on the command line.
- You can enter command line parameters in a file (the `PARFILE` parameter is excluded because parameter files cannot be nested).
- In interactive-command mode, the current job continues running, but logging to the terminal is stopped and the Import prompt is displayed. You can, for example, attach additional jobs to an executing or stopped job.



Data Pump Import: Transformations

You can remap:

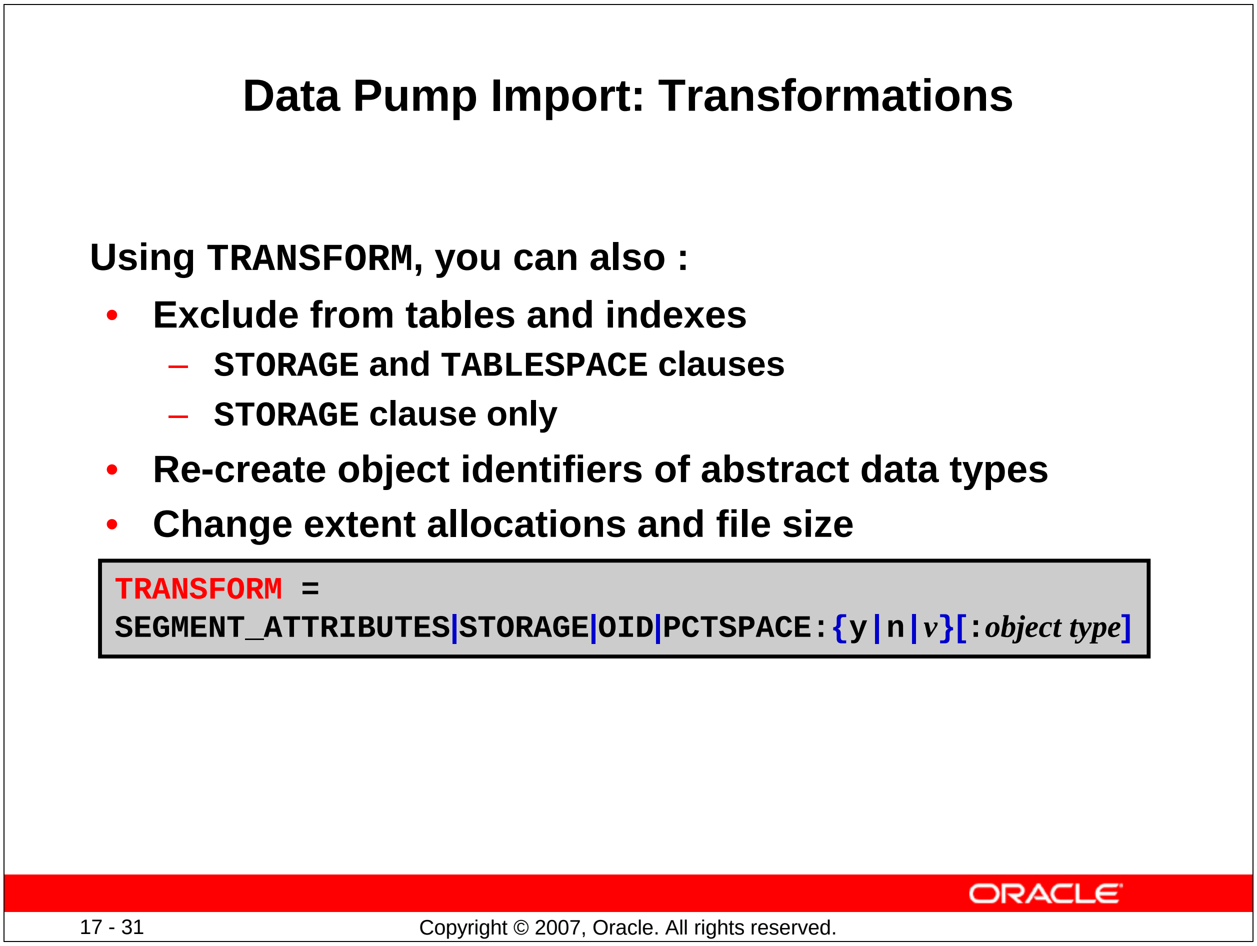
- Data files by using **REMAP_DATAFILE**
- Tablespaces by using **REMAP_TABLESPACE**
- Schemas by using **REMAP_SCHEMA**

```
REMAP_DATAFILE = 'C:\oradata\tbs6.f':'/u1/tbs6.f'
```

Data Pump Import: Transformations

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

- `REMAP_DATAFILE` is useful when moving databases across platforms that have different file-system semantics.
- `REMAP_TABLESPACE` enables objects to be moved from one tablespace to another.
- `REMAP_SCHEMA` provides the old `FROMUSER` /`TOUSER` capability to change object ownership.



Data Pump Import: Transformations

Using `TRANSFORM`, you can also :

- Exclude from tables and indexes
  - `STORAGE` and `TABLESPACE` clauses
  - `STORAGE` clause only
- Re-create object identifiers of abstract data types
- Change extent allocations and file size

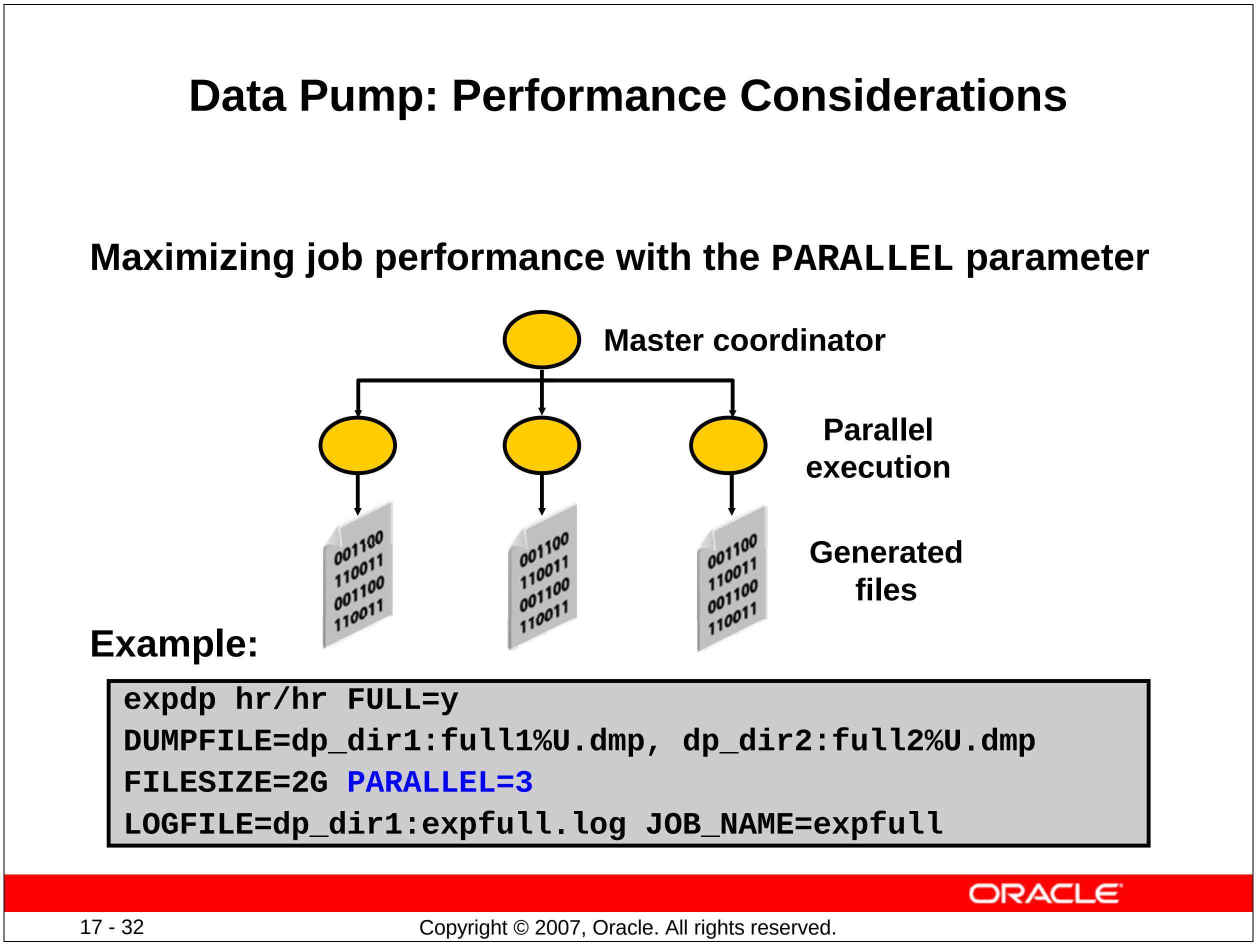
```
TRANSFORM =
SEGMENT_ATTRIBUTES|STORAGE|OID|PCTSPACE:{y|n|v}[:object type]
```

17 - 31

Data Pump Import: Transformations (continued)

The `TRANSFORM` parameter enables you to alter the object-creation DDL for specific objects or for all applicable objects being loaded. Specify the `TRANSFORM` parameter as shown in the slide. Note the following possible options:

- **`SEGMENT_ATTRIBUTES`:** If the value is specified as `Y`, segment attributes (physical attributes, storage attributes, tablespaces, and logging) are included.
- **`STORAGE`:** If the value is specified as `Y`, the `STORAGE` clauses are included.
- **`OID`:** Determines whether the object ID (OID) of abstract data types is reused or created as new. If the value is specified as `N`, the generation of the export OID clause for object types is suppressed. This is useful when you need to duplicate schemas across databases by using export and import, but you cannot guarantee that the object types will have identical OID values in those databases.
- **`PCTSPACE`:** Reduces the amount of space that is required for tablespaces by performing a shrink operation on tablespace storage allocation. The value supplied for this transformation must be a number greater than zero. It represents the percentage multiplier that is used to alter extent allocations and the size of data files.



Data Pump: Performance Considerations

Maximizing job performance with the PARALLEL parameter

Master coordinator

Parallel execution

Generated files

Example:

```
expdp hr/hr FULL=y
DUMPFILE=dp_dir1:full1%U.dmp, dp_dir2:full2%U.dmp
FILESIZE=2G PARALLEL=3
LOGFILE=dp_dir1:expfull.log JOB_NAME=expfull
```

Data Pump: Performance Considerations

You can improve job throughput with the `PARALLEL` parameter. The parallelism setting is enforced by the master process, which allocates work to be executed to worker processes that perform the data and metadata processing in an operation. These worker processes operate in parallel. In general, the degree of parallelism should be set to more than twice the number of CPUs on an instance. To maximize parallelism, you must supply at least one file for each degree of parallelism. If there are not enough dump files, the performance will not be optimal because multiple threads of execution will try to access the same dump file. The degree of parallelism can be reset at any time during a job.

The example in the slide shows a full database export. All data and metadata in the database will be exported. Dump files (`full101.dmp`, `full201.dmp`, `full102`.dmp, and so on) will be created in a round-robin fashion in the directories pointed to by the `dp_dir1` and `dp_dir2` directory objects. For best performance, these should be on separate I/O channels. Each file will be up to 2 GB in size (as necessary). Up to three files will be created initially, and more files will be created if needed. The job and master table have the same name: `expfull`. The log file will be written to `expfull.log` in the `dp_dir1` directory.

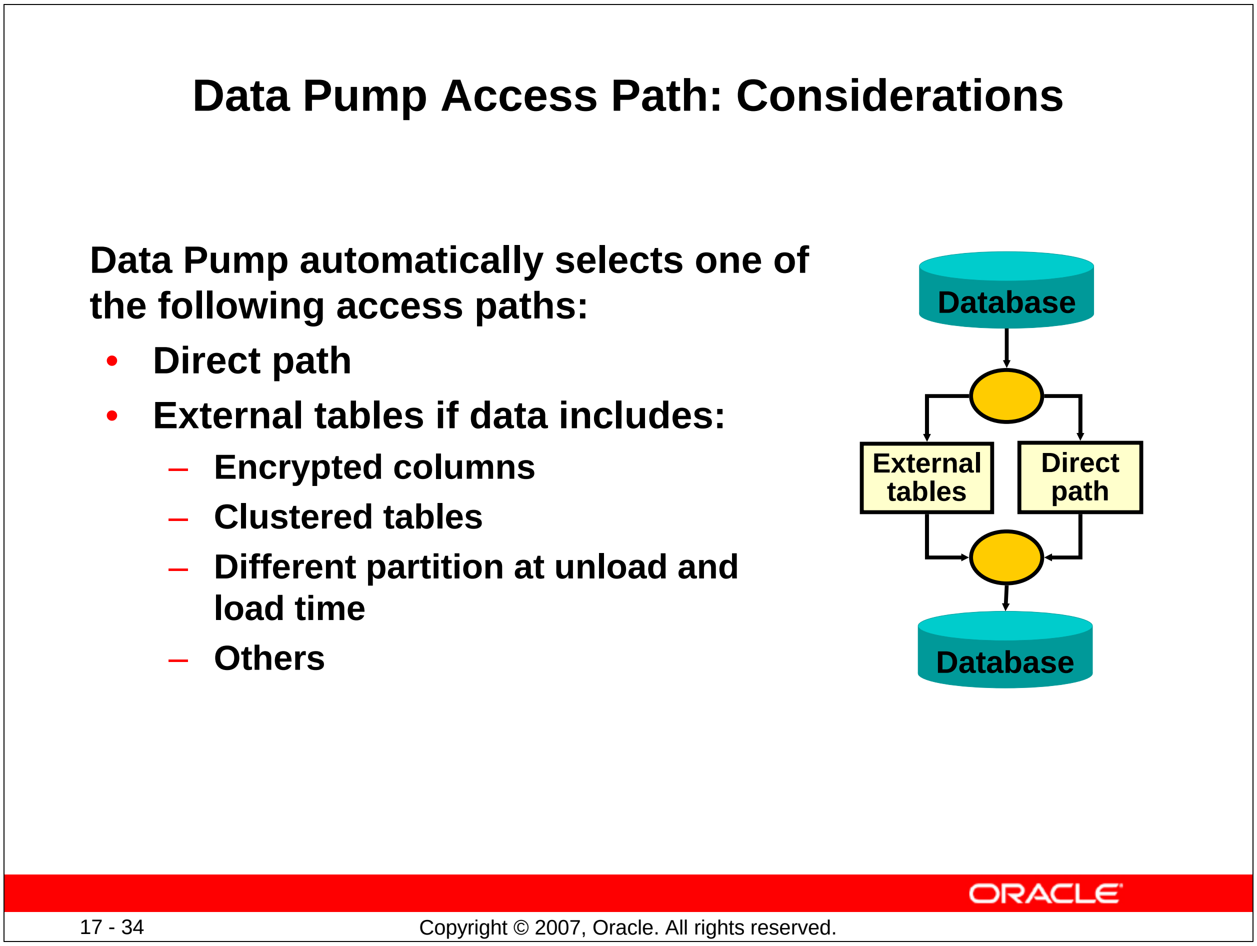
Performance Initialization Parameters

- Data Pump performance can be affected by:
  - `DISK_ASYNCH_IO`
  - `DB_BLOCK_CHECKING`
  - `DB_BLOCK_CHECKSUM`
- Set the following high to enable maximum parallelism:
  - `PROCESSES`
  - `SESSIONS`
  - `PARALLEL_MAX_SERVERS`
- Size generously:
  - Shared pool
  - Undo tablespace

Performance Initialization Parameters

You can try using the parameters listed in the slide to improve performance, although the results may not be the same on all platforms.

Additionally, the `SHARED_POOL_SIZE` and `UNDO_TABLESPACE` initialization parameters should be generously sized. The exact values will depend upon the size of your database. Turning off `DB_BLOCK_CHECKING` and `DB_BLOCK_CHECKSUM` only to improve Data Pump performance is not recommended because this will affect the detection of block corruption.



Data Pump Access Path: Considerations

Data Pump automatically selects one of the following access paths:

- Direct path
- External tables if data includes:
  - Encrypted columns
  - Clustered tables
  - Different partition at unload and load time
  - Others

Data Pump Direct Path: Considerations

Data Pump automatically selects the appropriate access method for each table.

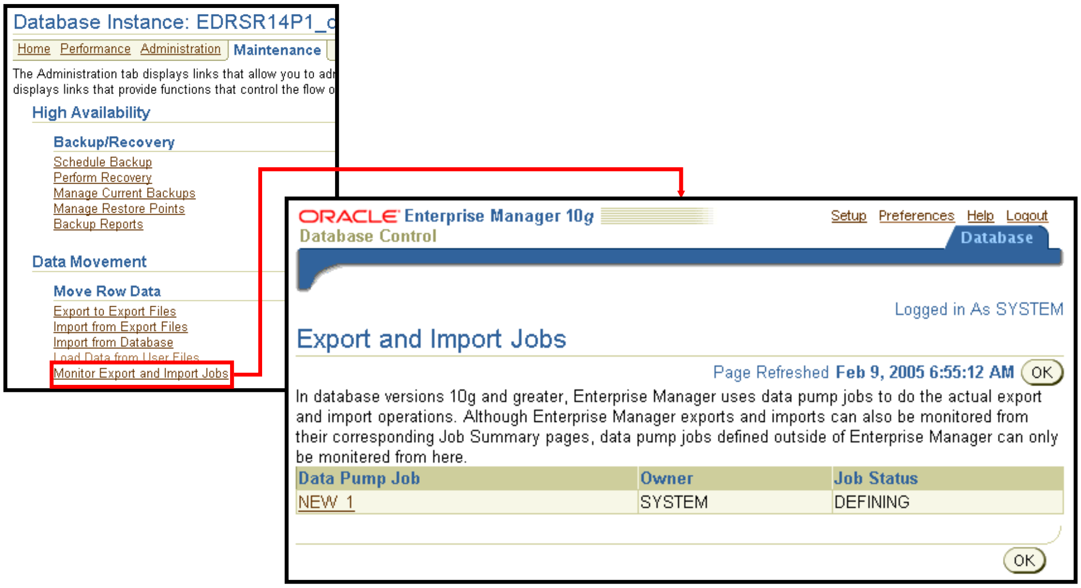
Direct path: Data Pump uses direct path load and unload when a table’s structure allows it and when maximum single-stream performance is desired.

External tables: Data Pump uses external tables for any of the following conditions:

- Tables with fine-grained access control enabled in insert and select modes
- Domain index for a LOB column
- Tables with active triggers defined
- Global index on partitioned tables with a single-partition load
- `BFILE` or opaque type columns
- Referential integrity constraint
- `VARRAY` columns with an embedded opaque type

Note: Because both methods support the same external data representation, data that is unloaded with one method can be loaded using the other method.

Using Enterprise Manager to Monitor Data Pump Jobs



Database Instance: EDRSR14P1_c

The Administration tab displays links that allow you to adr
displays links that provide functions that control the flow o

High Availability

Backup/Recovery

Data Movement

Move Row Data

Logged in As SYSTEM

Export and Import Jobs

Page Refreshed **Feb 9, 2005 6:55:12 AM** (OK)

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

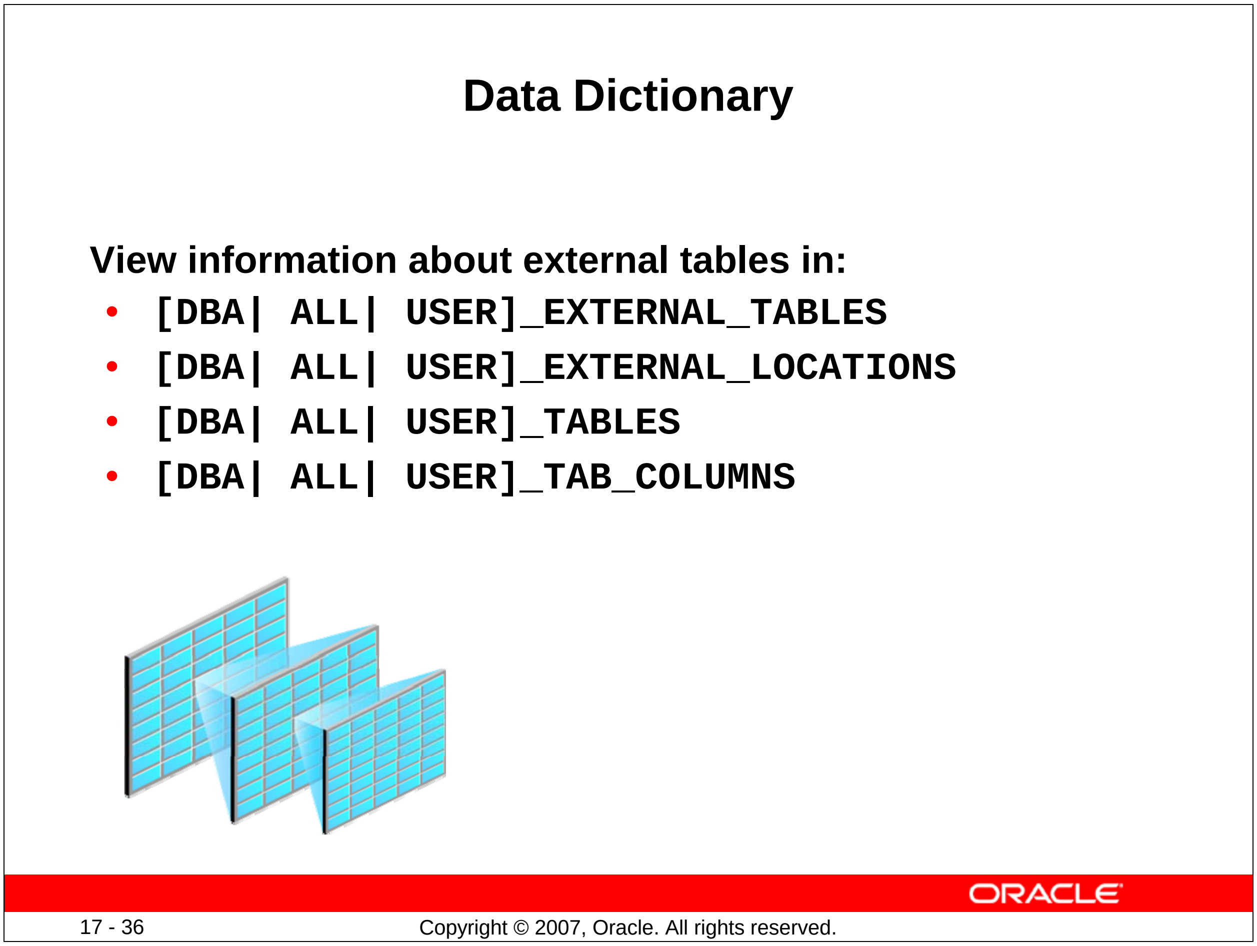
Data Pump Job	Owner	Job Status
NEW_1	SYSTEM	DEFINING

Using Enterprise Manager to Monitor Data Pump Jobs

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

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

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



Data Dictionary

View information about external tables in:

- **`[DBA| ALL| USER]_EXTERNAL_TABLES`**
- **`[DBA| ALL| USER]_EXTERNAL_LOCATIONS`**
- **`[DBA| ALL| USER]_TABLES`**
- **`[DBA| ALL| USER]_TAB_COLUMNS`**

Data Dictionary

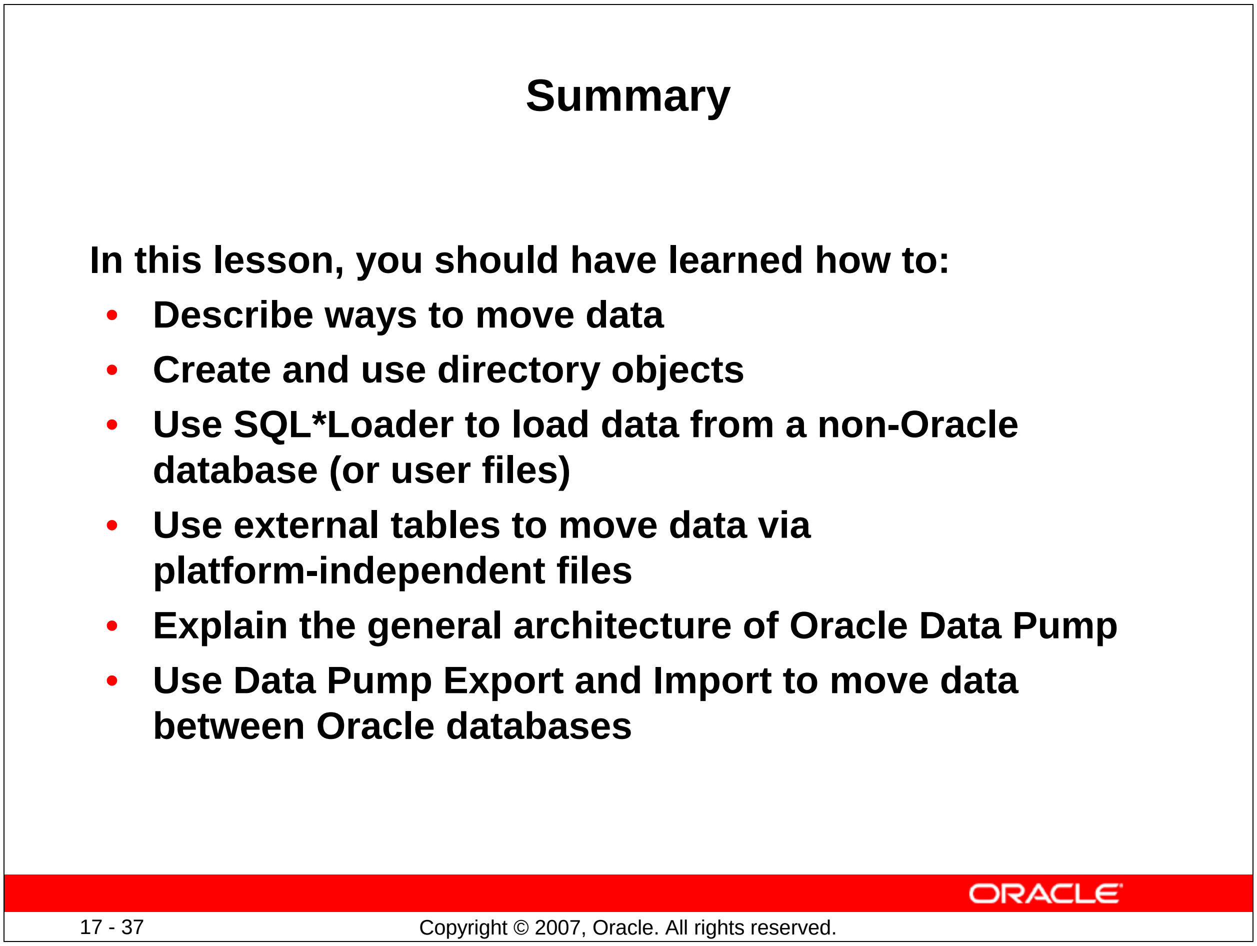
The data dictionary views in the slide list the following table information:

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

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

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

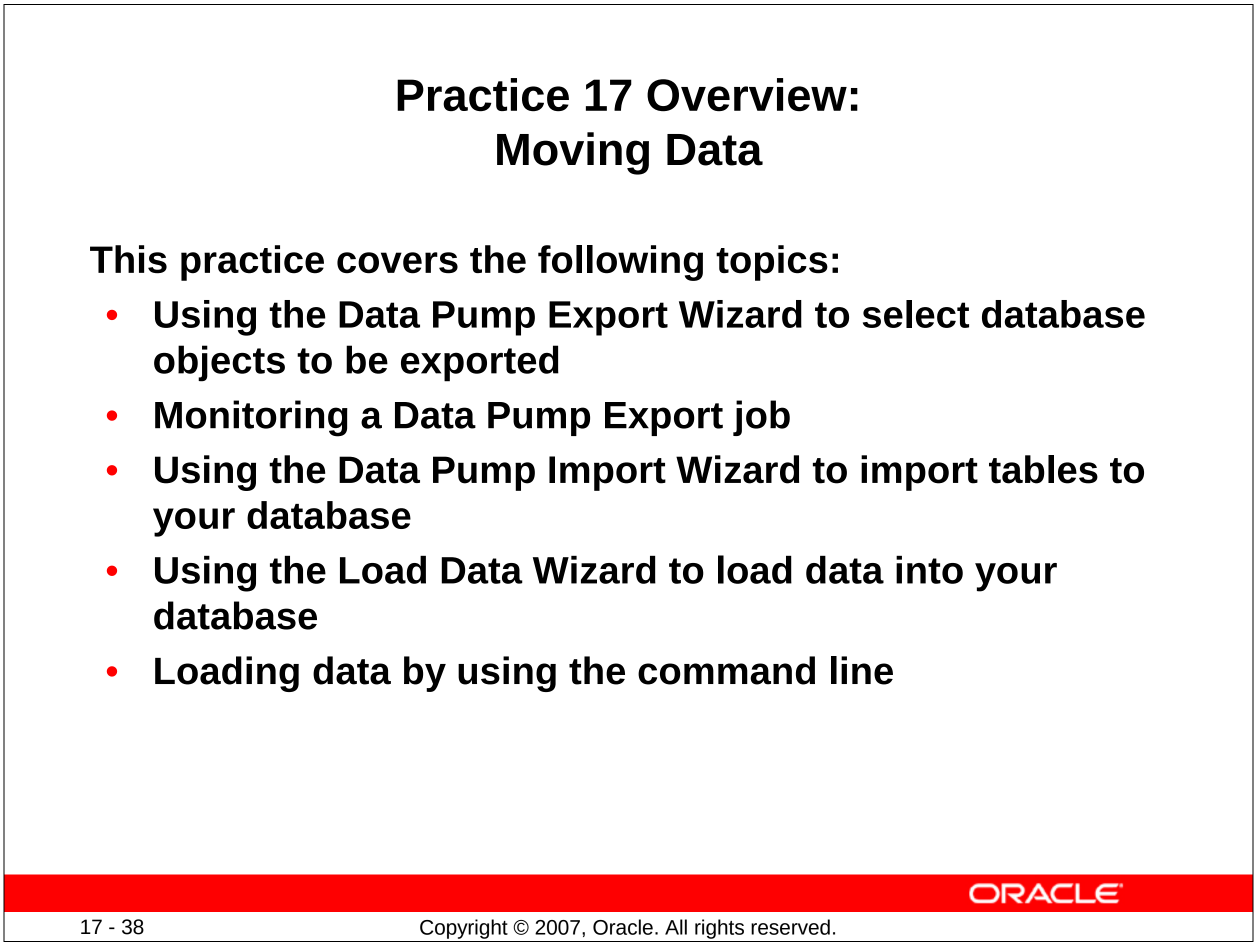
`[DBA| ALL| USER]_TAB_COLUMNS:` Descriptions of the columns of tables, views, and clusters in the database



Summary

In this lesson, you should have learned how to:

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



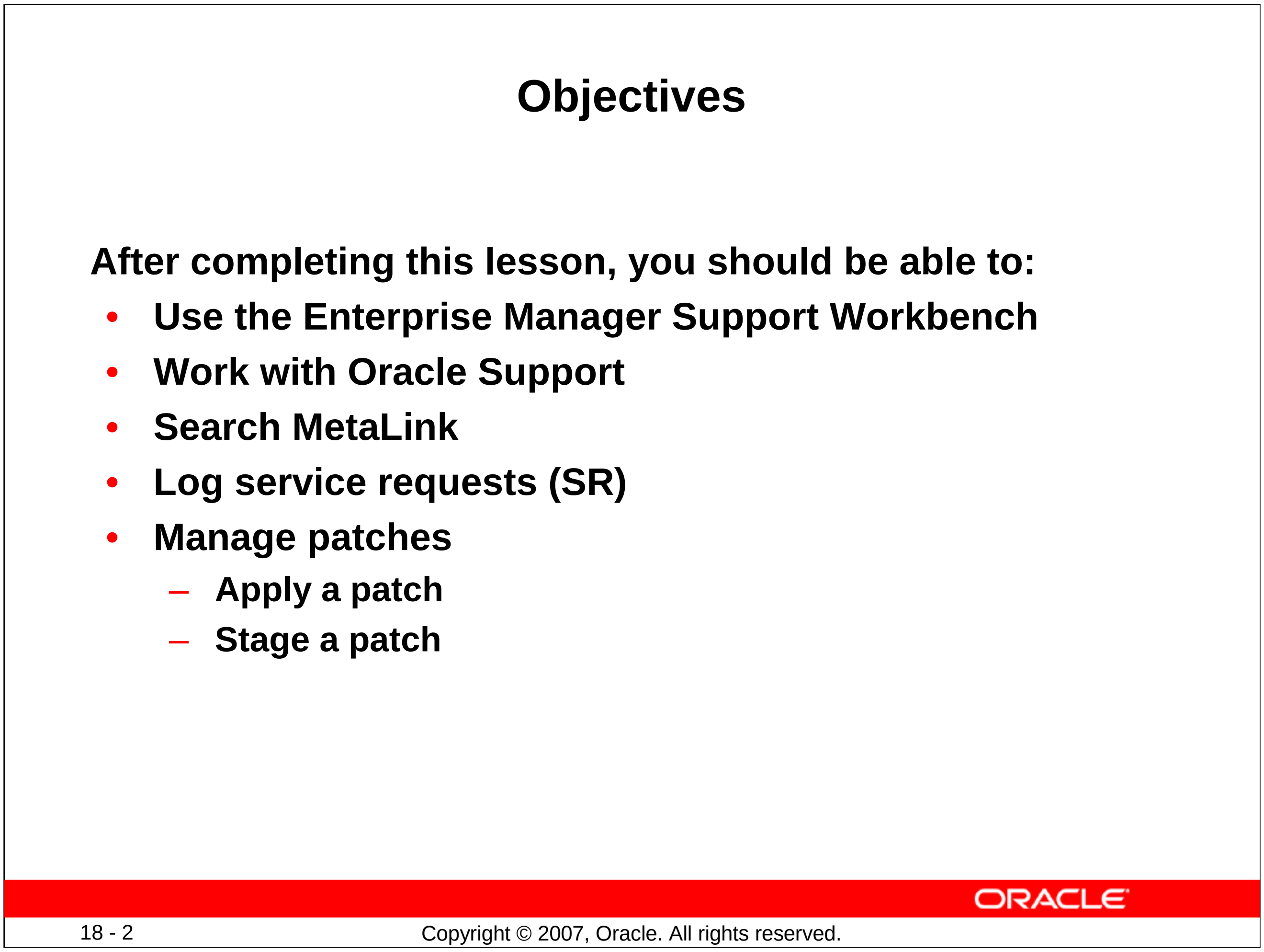
Practice 17 Overview: Moving Data

This practice covers the following topics:

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

18

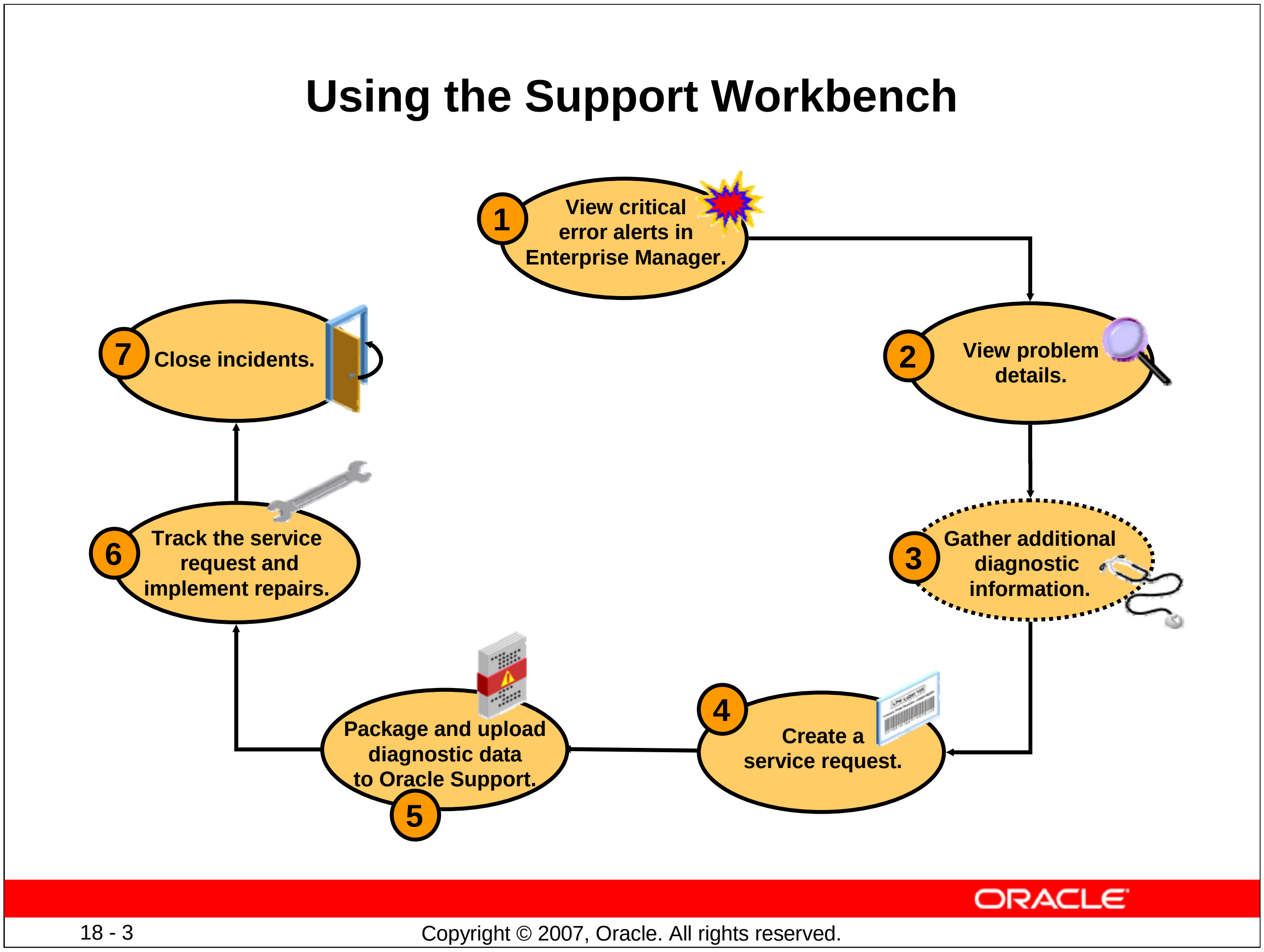
Enhancing Database Capabilities



Objectives

After completing this lesson, you should be able to:

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



Using the Support Workbench

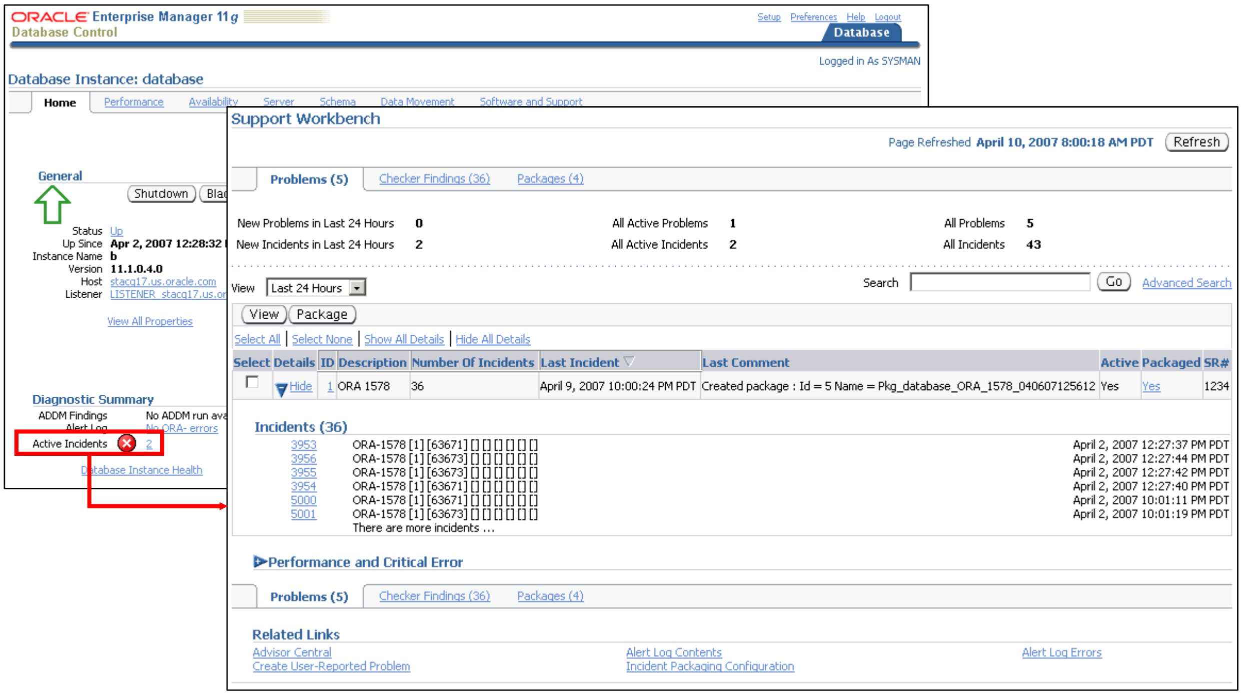
1. View critical error alerts in Enterprise Manager.
2. View problem details.
3. Gather additional diagnostic information.
4. Create a service request.
5. Package and upload diagnostic data to Oracle Support.
6. Track the service request and implement repairs.
7. Close incidents.

Using the Support Workbench

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

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

Viewing Critical Error Alerts in Enterprise Manager



Database Instance: database

General

Status	Up
Up Since	Apr 2, 2007 12:28:32
Instance Name	b
Version	11.1.0.4.0
Host	stacq17.us.oracle.com
Listener	LISTENER_stacq17.us.or

Diagnostic Summary

ADDM Findings	No ADDM run ava
Alert Log	No ORA- errors
Active Incidents	☒ 2

Support Workbench

Page Refreshed **April 10, 2007 8:00:18 AM PDT**

Problems (5) | Checker Findings (36) | Packages (4)

New Problems in Last 24 Hours	0	All Active Problems	1	All Problems	5
New Incidents in Last 24 Hours	2	All Active Incidents	2	All Incidents	43

View: Last 24 Hours

Search

Select	Details	ID	Description	Number Of Incidents	Last Incident ▽	Last Comment	Active	Packaged	SR#
□	▼Hide	1	ORA 1578	36	April 9, 2007 10:00:24 PM PDT	Created package : Id = 5 Name = Pkg_database_ORA_1578_040607125612	Yes	Yes	1234

Incidents (36)

3953	ORA-1578 [1] [63671] [] [] [] [] [] []	April 2, 2007 12:27:37 PM PDT
3956	ORA-1578 [1] [63673] [] [] [] [] [] []	April 2, 2007 12:27:44 PM PDT
3955	ORA-1578 [1] [63673] [] [] [] [] [] []	April 2, 2007 12:27:42 PM PDT
3954	ORA-1578 [1] [63671] [] [] [] [] [] []	April 2, 2007 12:27:40 PM PDT
5000	ORA-1578 [1] [63671] [] [] [] [] [] []	April 2, 2007 10:01:11 PM PDT
5001	ORA-1578 [1] [63673] [] [] [] [] [] []	April 2, 2007 10:01:19 PM PDT
	There are more incidents ...	

▷Performance and Critical Error

Problems (5) | Checker Findings (36) | Packages (4)

Related Links

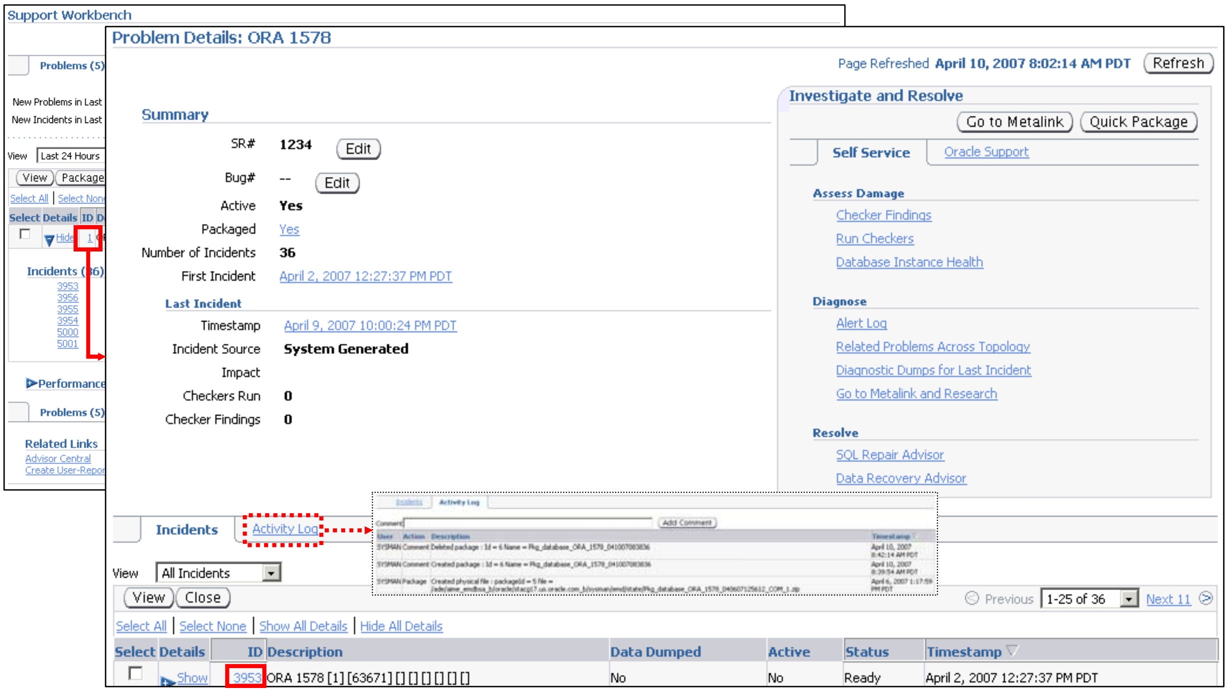
Viewing Critical Error Alerts in Enterprise Manager

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

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

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

Viewing Problem Details



Support Workbench

Problems (5)

New Problems in Last

New Incidents in Last

View: Last 24 Hours

View | Package

Select All | Select None

Select	Details	ID
□	Hide	1

Incidents (36)

3953
3956
3955
3954
5000
5001

Performance

Problems (5)

Related Links

Advisor Central

Create User-Repor

Problem Details: ORA 1578

Page Refreshed **April 10, 2007 8:02:14 AM PDT** Refresh

Summary

SR#	**1234** Edit
Bug#	-- Edit
Active	**Yes**
Packaged	Yes
Number of Incidents	**36**
First Incident	April 2, 2007 12:27:37 PM PDT

Last Incident

Timestamp	April 9, 2007 10:00:24 PM PDT
Incident Source	**System Generated**
Impact	
Checkers Run	**0**
Checker Findings	**0**

Investigate and Resolve

Go to Metalink | Quick Package

Self Service | Oracle Support

Assess Damage

- Checker Findings
- Run Checkers
- Database Instance Health

Diagnose

- Alert Log
- Related Problems Across Topology
- Diagnostic Dumps for Last Incident
- Go to Metalink and Research

Resolve

- SQL Repair Advisor
- Data Recovery Advisor

Incidents | Activity Log

Comment: Add Comment

User	Action	Description	Timestamp
SYSMAN	Comment	Deleted package : Id = 6 Name = Pkg_database_ORA_1578_041007083836	April 10, 2007 8:42:14 AM PDT
SYSMAN	Comment	Created package : Id = 6 Name = Pkg_database_ORA_1578_041007083836	April 10, 2007 8:39:54 AM PDT
SYSMAN	Package	Created physical file : packageId = 5 file = /ade/ame_emdbsa_b/oracle/stacg17.us.oracle.com_b/sysman/emd/state/Pkg_database_ORA_1578_040607125612_COM_1.zip	April 6, 2007 1:17:59 PM PDT

View: All Incidents

View | Close

Previous 1-25 of 36 Next 11

Select All | Select None | Show All Details | Hide All Details

Select	Details	ID	Description	Data Dumped	Active	Status	Timestamp
□	Show	3953	ORA 1578 [1] [63671] [] [] [] [] [] []	No	No	Ready	April 2, 2007 12:27:37 PM PDT

Viewing Problem Details

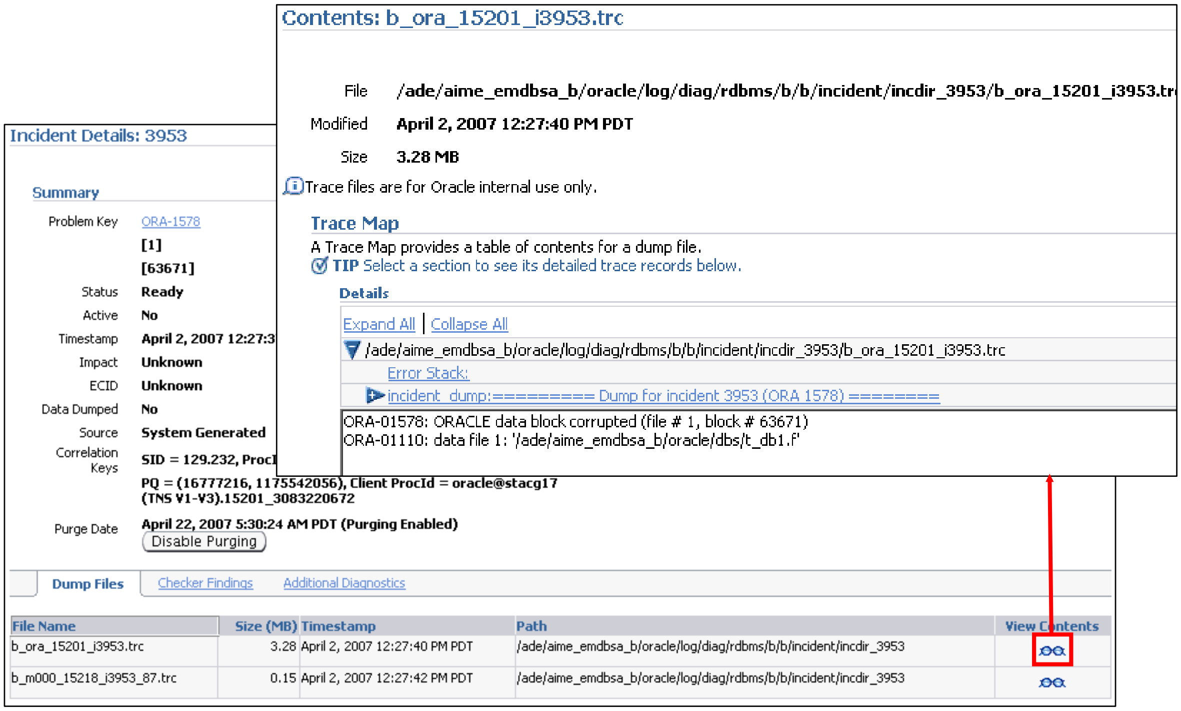
From the Problems subpage on the Support Workbench page, click the ID of the problem that you want to investigate. This takes you to the corresponding Problem Details page.

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

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

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

Viewing Incident Details: Dump Files



Contents: b_ora_15201_i3953.trc

File /ade/aime_emdbsa_b/oracle/log/diag/rdbms/b/b/incident/incdir_3953/b_ora_15201_i3953.tr

Modified **April 2, 2007 12:27:40 PM PDT**

Size **3.28 MB**

Trace files are for Oracle internal use only.

Trace Map

A Trace Map provides a table of contents for a dump file.

TIP Select a section to see its detailed trace records below.

Details

Expand All | Collapse All

▼ /ade/aime_emdbsa_b/oracle/log/diag/rdbms/b/b/incident/incdir_3953/b_ora_15201_i3953.trc

Error Stack:

▶ incident_dump:========= Dump for incident 3953 (ORA 1578) ========

```
ORA-01578: ORACLE data block corrupted (file # 1, block # 63671)
ORA-01110: data file 1: '/ade/aime_emdbsa_b/oracle/dbs/t_db1.f'
```

Incident Details: 3953

Summary

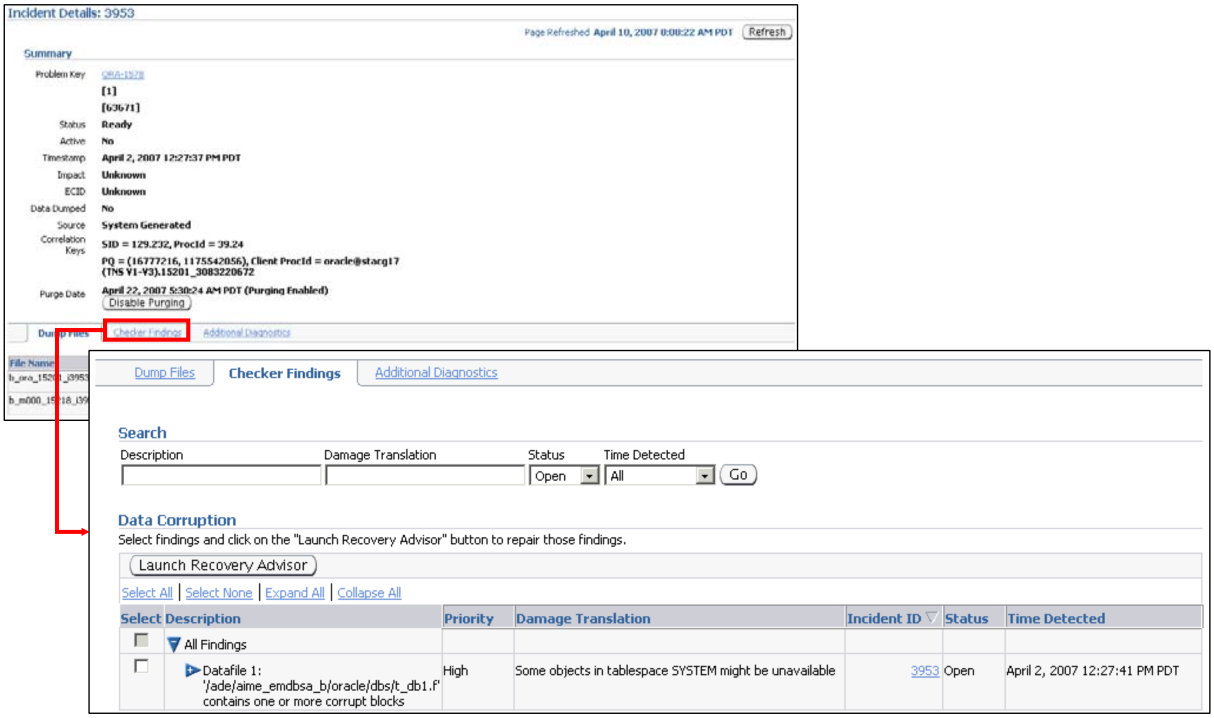
Problem Key	ORA-1578 **[1]** **[63671]**
Status	**Ready**
Active	**No**
Timestamp	**April 2, 2007 12:27:3**
Impact	**Unknown**
ECID	**Unknown**
Data Dumped	**No**
Source	**System Generated**
Correlation Keys	**SID = 129.232, ProcI** **PQ = (16777216, 1175542056), Client ProcId = oracle@stacg17** **(TNS V1-V3).15201_3083220672**
Purge Date	**April 22, 2007 5:30:24 AM PDT (Purging Enabled)** Disable Purging

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

Viewing Incident Details: Dump Files

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

Viewing Incident Details: Checker Findings



Incident Details: 3953

Page Refreshed April 10, 2007 0:00:22 AM PDT

Summary

Problem Key	ORA-1578 [1] [63671]
Status	Ready
Active	No
Timestamp	April 2, 2007 12:27:37 PM PDT
Impact	Unknown
ECID	Unknown
Data Dumped	No
Source	System Generated

Correlation Keys	SID = 129.232, ProcId = 39.24
	PQ = (16777216, 1175542056), Client ProcId = oracle@stacg17 (TNS V1-V3).15201_3083220672

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

Search

Description	Damage Translation	Status	Time Detected
		Open	All

Data Corruption

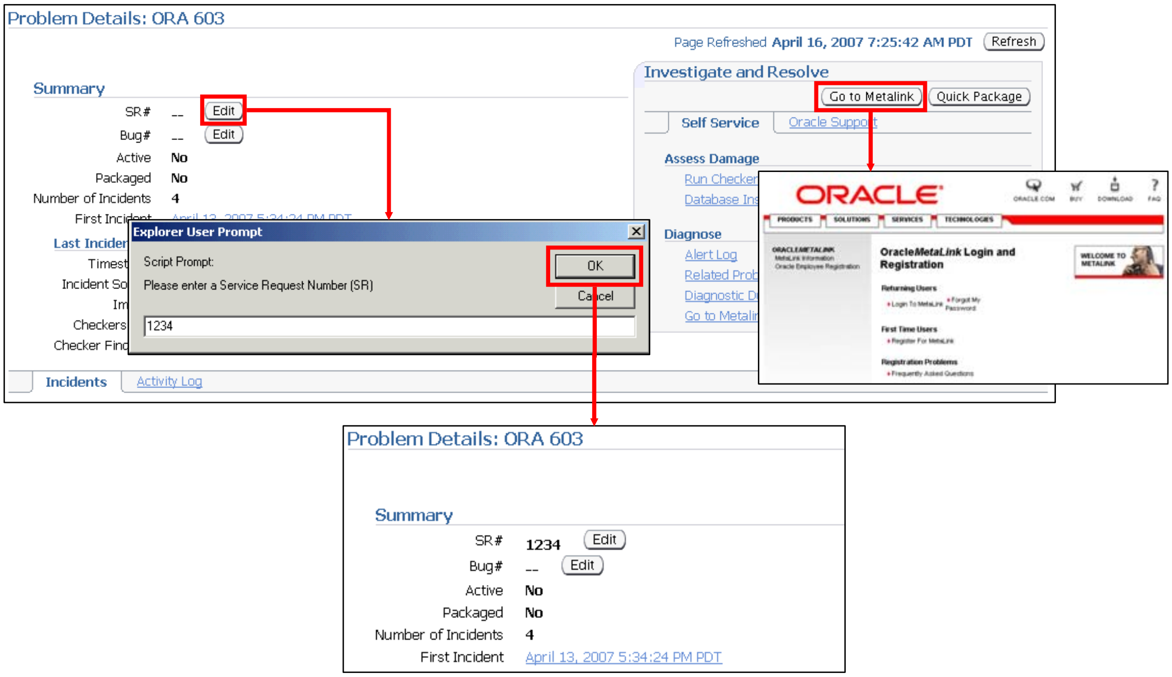
Select findings and click on the "Launch Recovery Advisor" button to repair those findings.

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

Viewing Incident Details: Checker Findings

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

Creating a Service Request



Problem Details: ORA 603

Page Refreshed April 16, 2007 7:25:42 AM PDT

Summary

SR# --
Bug# --
Active No
Packaged No
Number of Incidents 4

Explorer User Prompt

Script Prompt:
Please enter a Service Request Number (SR)

1234

Problem Details: ORA 603

Summary

SR# 1234
Bug# --
Active No
Packaged No
Number of Incidents 4
First Incident April 13, 2007 5:34:24 PM PDT

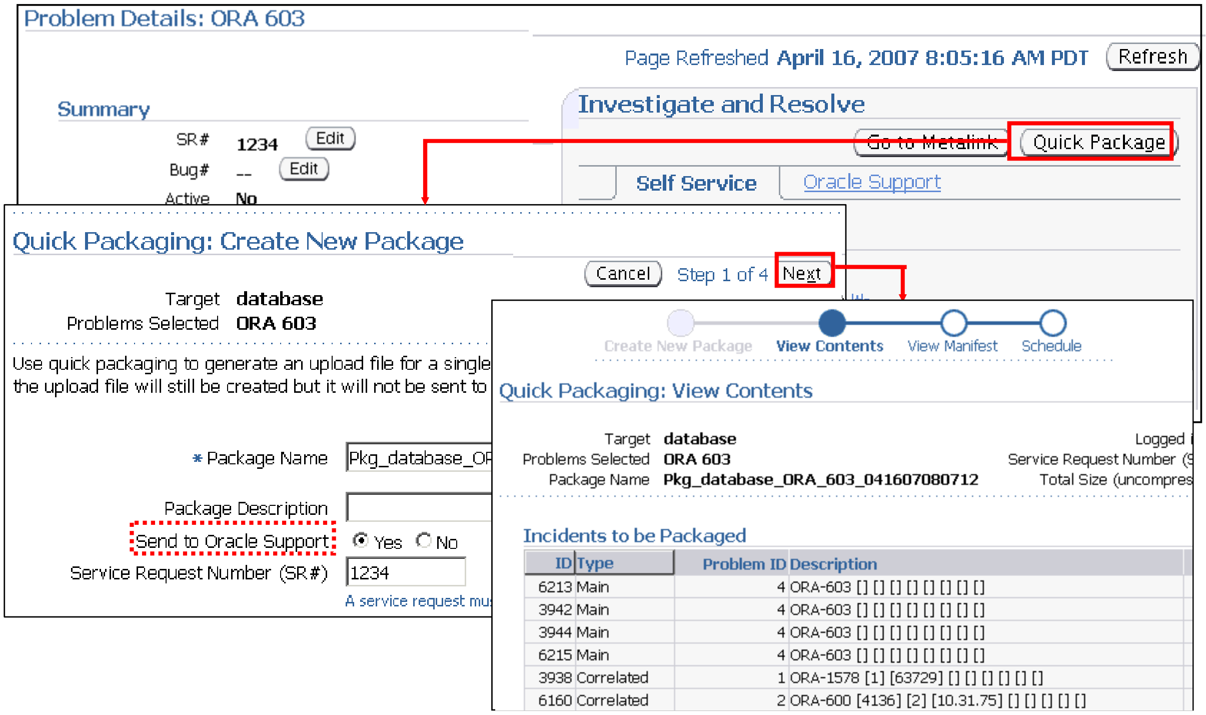
Creating a Service Request

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

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

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

Packaging and Uploading Diagnostic Data to Oracle Support



Problem Details: ORA 603

Page Refreshed **April 16, 2007 8:05:16 AM PDT** (Refresh)

Summary

SR#	1234	(Edit)
Bug#	--	(Edit)
Active	No	

Investigate and Resolve

Quick Packaging: Create New Package

(Cancel) Step 1 of 4 (Next)

Target **database**
Problems Selected **ORA 603**

Use quick packaging to generate an upload file for a single
the upload file will still be created but it will not be sent to

* Package Name: Pkg_database_OF

Package Description:

Send to Oracle Support: ● Yes ○ No

Service Request Number (SR#): 1234

A service request mu

Quick Packaging: View Contents

Target **database**
Problems Selected **ORA 603**
Package Name **Pkg_database_ORA_603_041607080712**

Logged
Service Request Number (S
Total Size (uncompres

Incidents to be Packaged

ID	Type	Problem ID	Description
6213	Main	4	ORA-603 [] [] [] [] [] [] [] []
3942	Main	4	ORA-603 [] [] [] [] [] [] [] []
3944	Main	4	ORA-603 [] [] [] [] [] [] [] []
6215	Main	4	ORA-603 [] [] [] [] [] [] [] []
3938	Correlated	1	ORA-1578 [1] [63729] [] [] [] [] [] []
6160	Correlated	2	ORA-600 [4136] [2] [10.31.75] [] [] [] [] []

Packaging and Uploading Diagnostic Data to Oracle Support

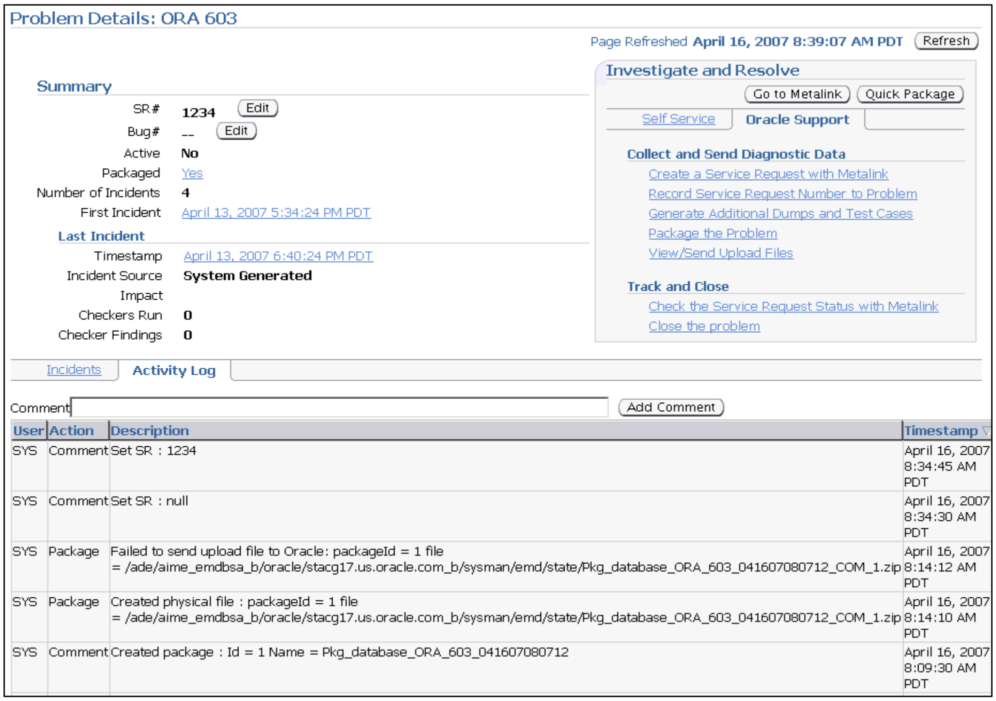
The Support Workbench provides two methods for creating and uploading an incident package: the Quick Packaging method and the Advanced Packaging method. The example in the slide shows how to use Quick Packaging.

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

To package and upload diagnostic data to Oracle Support:

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

Tracking the Service Request and Implementing Repairs



Problem Details: ORA 603

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

Summary

SR# 1234 Edit
Bug# -- Edit
Active No
Packaged Yes
Number of Incidents 4
First Incident April 13, 2007 5:34:24 PM PDT

Last Incident

Timestamp April 13, 2007 6:40:24 PM PDT
Incident Source System Generated
Impact
Checkers Run 0
Checker Findings 0

Investigate and Resolve

Collect and Send Diagnostic Data

- Create a Service Request with Metalink
- Record Service Request Number to Problem
- Generate Additional Dumps and Test Cases
- Package the Problem
- View/Send Upload Files

Track and Close

- Check the Service Request Status with Metalink
- Close the problem

Comment Add Comment

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

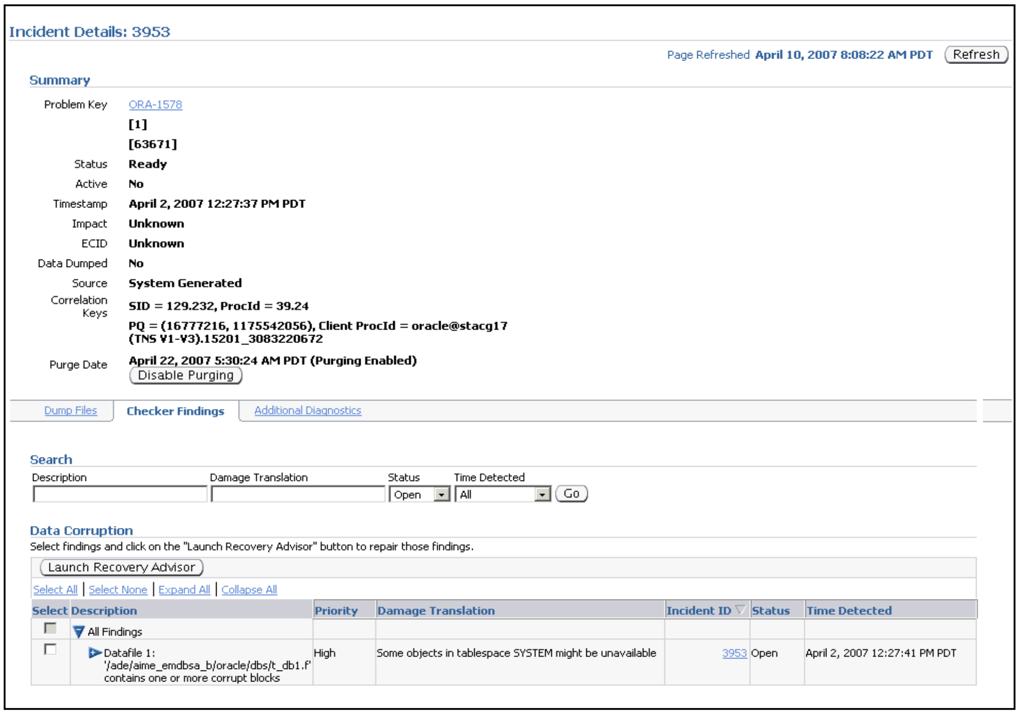
Tracking the Service Request and Implementing Repairs

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

- Add an Oracle bug number to the problem information. On the Problem Details page, click the Edit button that is adjacent to the Bug# label. This is for your reference only.
- Add comments to the problem activity log:
  1. Access the Problem Details page for the problem.
  2. Click Activity Log to display the Activity Log subpage.
  3. In the Comment field, enter a comment and then click Add Comment.

     Your comment is recorded in the activity log.
- Respond to a request by Oracle Support to provide additional diagnostics. Your Oracle Support representative can provide instructions for gathering and uploading additional diagnostics.

Tracking the Service Request and Implementing Repairs



Incident Details: 3953

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

Summary

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

Dump Files | **Checker Findings** | Additional Diagnostics

Search

Description [ ] Damage Translation [ ] Status [Open] Time Detected [All] (Go)

Data Corruption

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

(Launch Recovery Advisor)

Select All | Select None | Expand All | Collapse All

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

Tracking the Service Request and Implementing Repairs (continued)

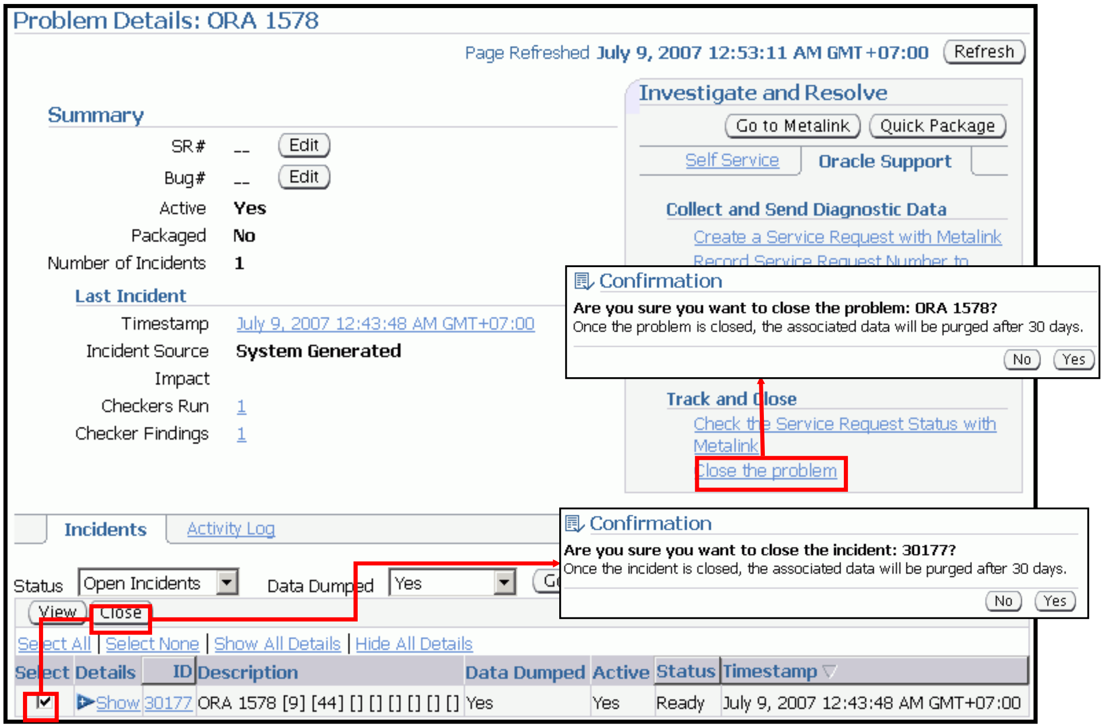
From the Incident Details page, you can run an Oracle advisor to implement repairs. Access the suggested advisor in one of the following places:

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

The advisors that help you repair critical errors are:

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

Closing Incidents and Problems



Problem Details: ORA 1578

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

Summary

SR# -- (Edit)

Bug# -- (Edit)

Active **Yes**

Packaged **No**

Number of Incidents **1**

Last Incident

Timestamp July 9, 2007 12:43:48 AM GMT+07:00

Incident Source **System Generated**

Impact

Checkers Run 1

Checker Findings 1

Investigate and Resolve

(Go to Metalink) (Quick Package)

Self Service | **Oracle Support**

Collect and Send Diagnostic Data

Create a Service Request with Metalink

Record Service Request Number to

Track and Close

Check the Service Request Status with Metalink

Close the problem

Confirmation

Are you sure you want to close the problem: ORA 1578?

Once the problem is closed, the associated data will be purged after 30 days.

(No) (Yes)

Incidents | Activity Log

Status: Open Incidents ▾ Data Dumped: Yes ▾ (Go)

(View) (Close)

Select All | Select None | Show All Details | Hide All Details

Select	Details	ID	Description	Data Dumped	Active	Status	Timestamp ▽
☑	▶Show	30177	ORA 1578 [9] [44] [] [] [] [] [] []	Yes	Yes	Ready	July 9, 2007 12:43:48 AM GMT+07:00

Confirmation

Are you sure you want to close the incident: 30177?

Once the incident is closed, the associated data will be purged after 30 days.

(No) (Yes)

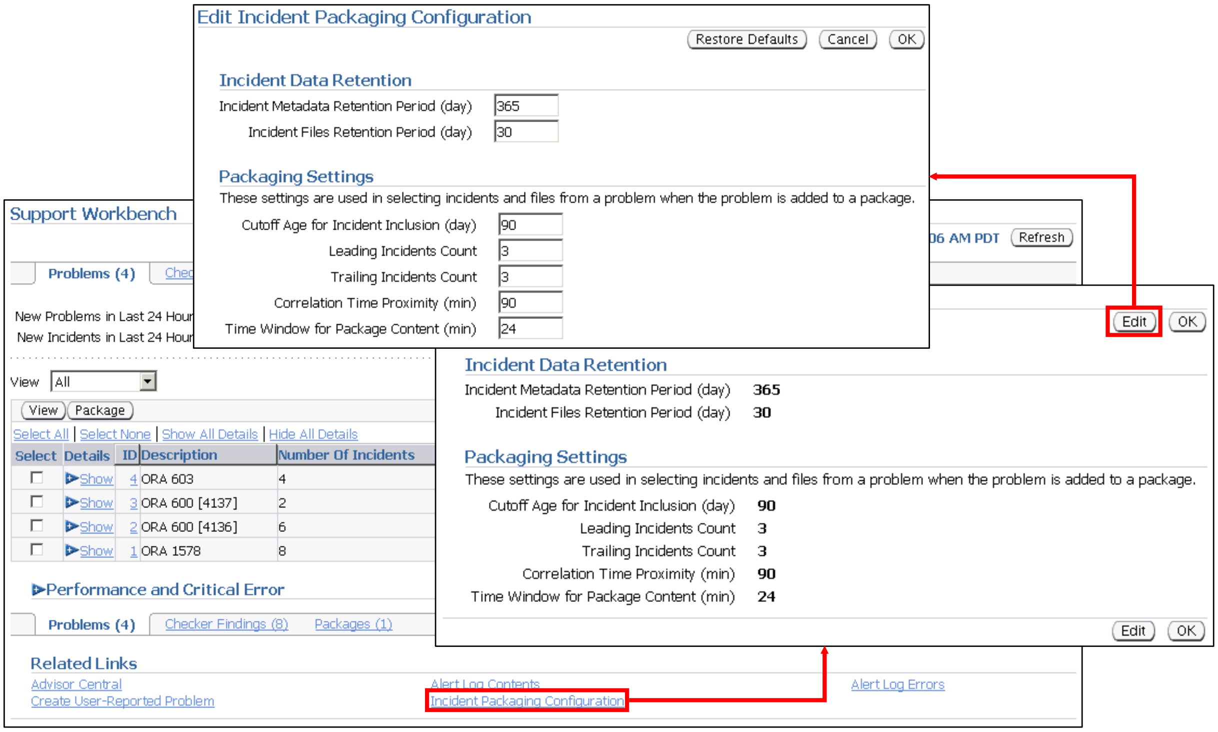
Closing Incidents and Problems

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

To close incidents:

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

Incident Packaging Configuration



Edit Incident Packaging Configuration

Incident Data Retention

Incident Metadata Retention Period (day) 365

Incident Files Retention Period (day) 30

Packaging Settings

These settings are used in selecting incidents and files from a problem when the problem is added to a package.

Cutoff Age for Incident Inclusion (day) 90

Leading Incidents Count 3

Trailing Incidents Count 3

Correlation Time Proximity (min) 90

Time Window for Package Content (min) 24

Support Workbench

New Problems in Last 24 Hour

New Incidents in Last 24 Hour

View All

View
Package
Select All | Select None | Show All Details | Hide All Details

Select	Details	ID	Description	Number Of Incidents
□	▶Show	4	ORA 603	4
□	▶Show	3	ORA 600 [4137]	2
□	▶Show	2	ORA 600 [4136]	6
□	▶Show	1	ORA 1578	8

Related Links

Advisor Central
Create User-Reported Problem
Alert Log Contents
Incident Packaging Configuration
Alert Log Errors
Edit
OK

Incident Data Retention

Incident Metadata Retention Period (day) 365

Incident Files Retention Period (day) 30

Packaging Settings

These settings are used in selecting incidents and files from a problem when the problem is added to a package.

Cutoff Age for Incident Inclusion (day) 90

Leading Incidents Count 3

Trailing Incidents Count 3

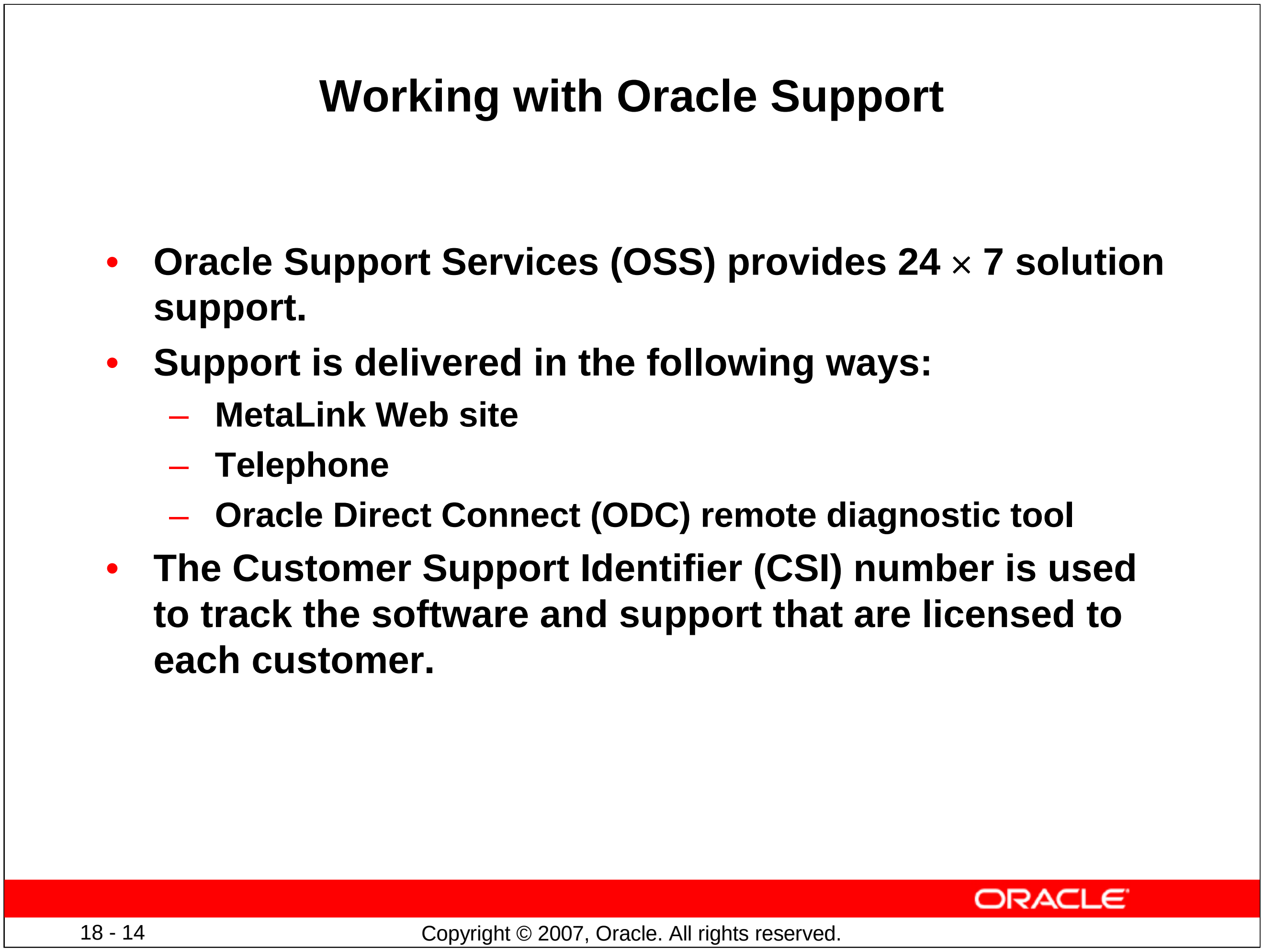
Correlation Time Proximity (min) 90

Time Window for Package Content (min) 24

Incident Packaging Configuration

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

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



Working with Oracle Support

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

ORACLE

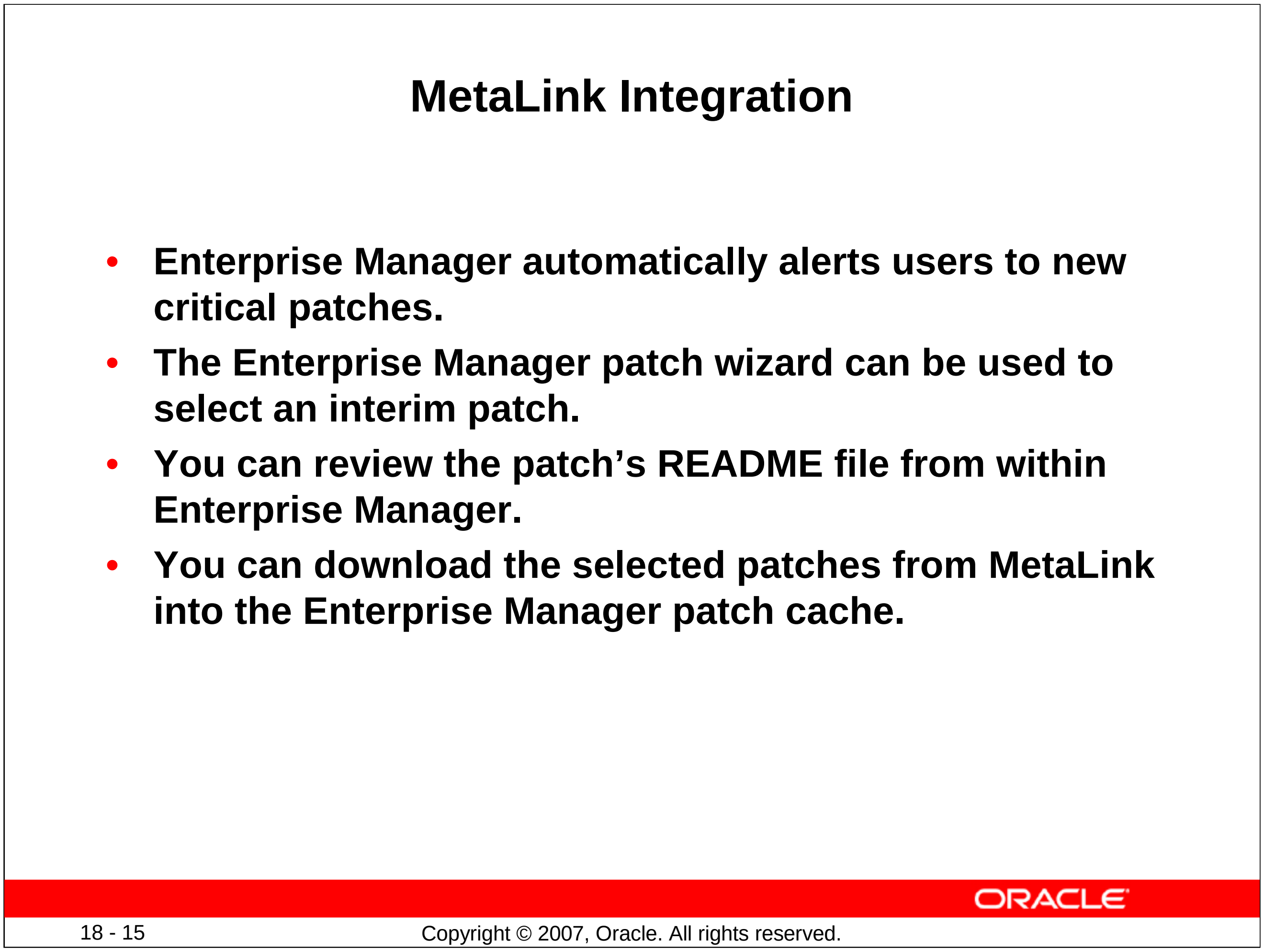
Working with Oracle Support

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

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

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

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



MetaLink Integration

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

ORACLE

18 - 15

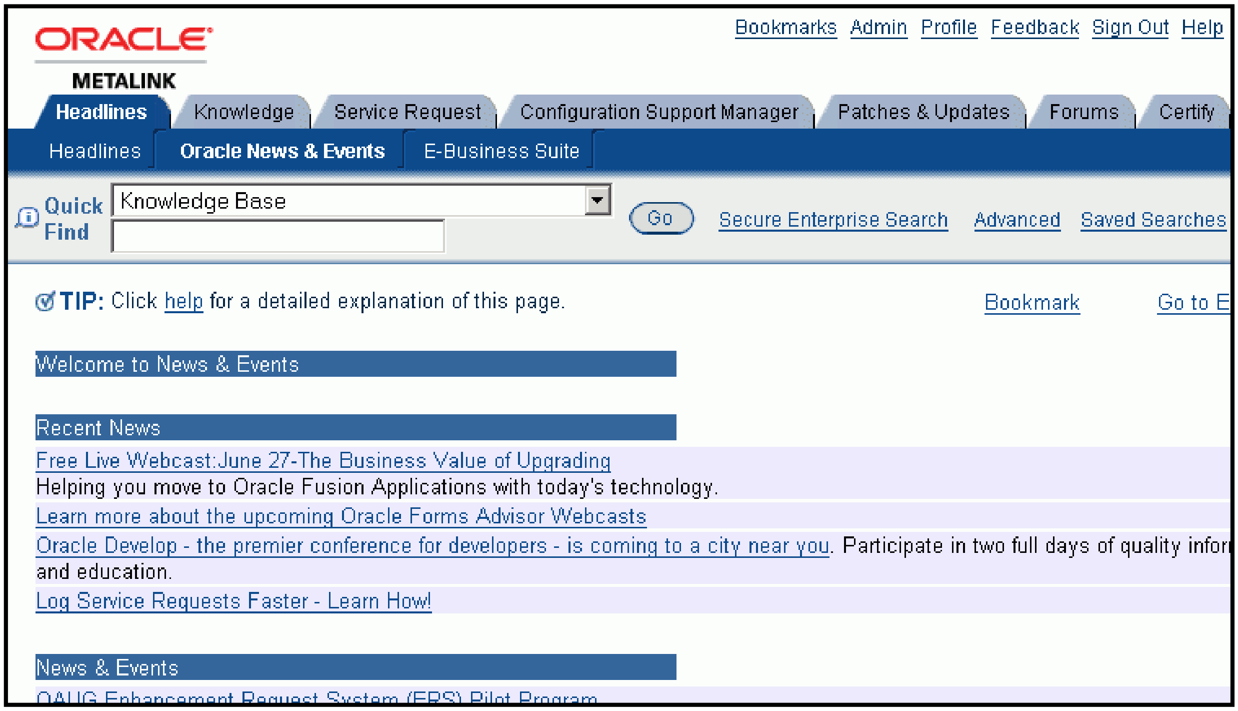
MetaLink Integration

Oracle Enterprise Manager (Enterprise Manager) significantly facilitates software patching with its built-in MetaLink integration. Enterprise Manager automatically alerts users to new critical patches and flags all systems that require a specific patch. You can invoke the Enterprise Manager patch wizard to determine what interim patches are available for installation. Alternatively, you can use the patch wizard to select an interim patch and determine whether any of your systems require that patch. You can review the patch details and README patch notes directly from within Enterprise Manager.

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

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

Using MetaLink



Quick Find: Knowledge Base

TIP: Click help for a detailed explanation of this page.

Welcome to News & Events

Recent News

Free Live Webcast:June 27-The Business Value of Upgrading
Helping you move to Oracle Fusion Applications with today's technology.

Learn more about the upcoming Oracle Forms Advisor Webcasts

Oracle Develop - the premier conference for developers - is coming to a city near you. Participate in two full days of quality infor and education.

Log Service Requests Faster - Learn How!

News & Events

OAUG Enhancement Request System (ERS) Pilot Program

Using MetaLink

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

MetaLink has a variety of tools and methods available for researching problems.

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

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

Using MetaLink (continued)

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

The following information is available on the Knowledge tab pages:

- User documentation
- Electronic technical reference manuals (eTRMs)
- Frequently asked questions (FAQs)
- Listing of educational offerings
- Self-service toolkits (SSTK)
- Business flows
- Resolution flows

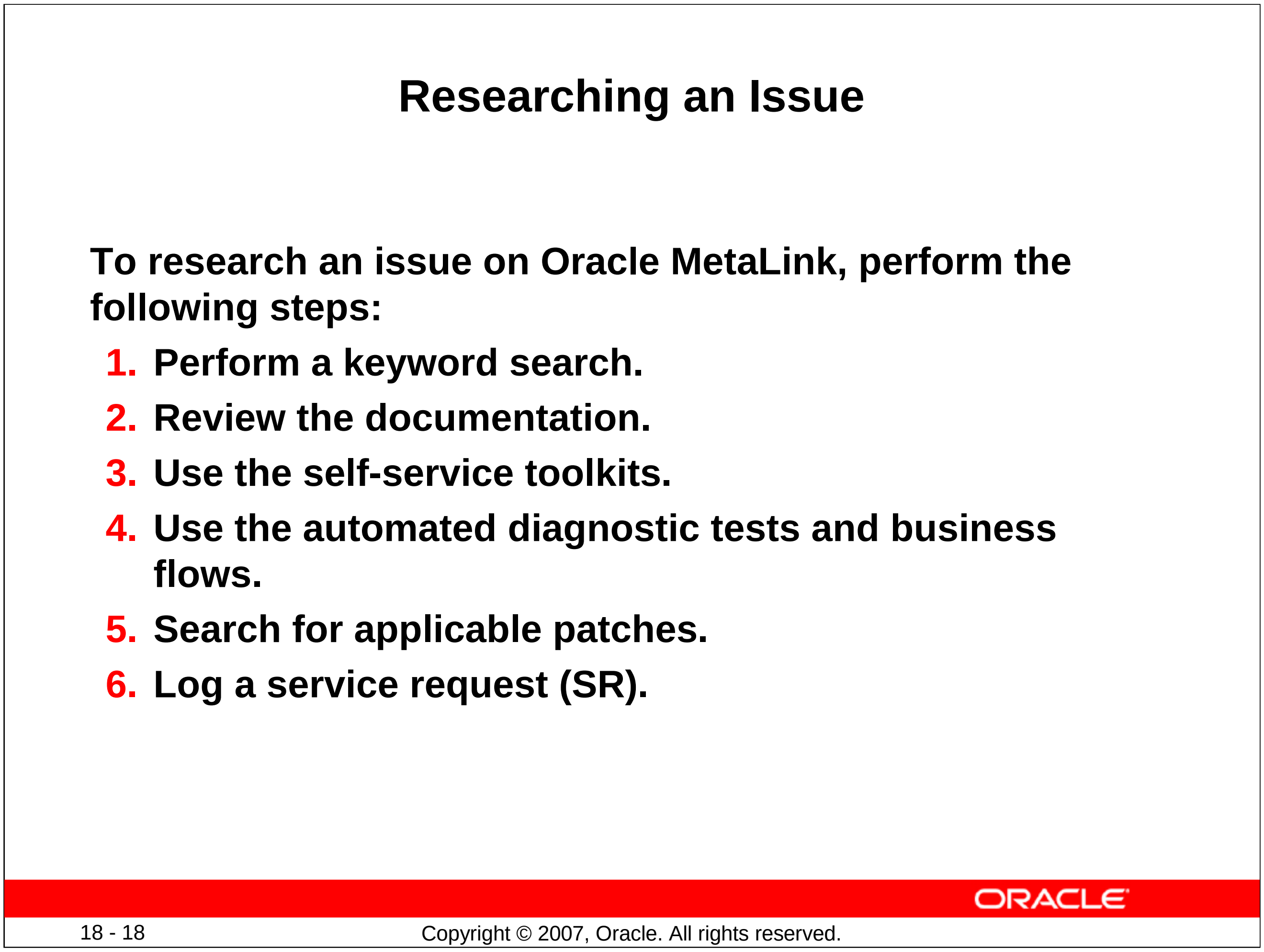
MetaLink Forums (Forums) enable you to interact with other Oracle customers to share ideas and discuss Oracle products. You can use MetaLink Forums to find out how other customers perform complex tasks or meet various business requirements with Oracle products. You should not use Forums as a substitute for logging an SR.

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

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

Note: For detailed information about performing these searches, refer to MetaLink Technical Note 166650.1 ("Working Effectively with Global Customer Support").

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



Researching an Issue

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

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

18 - 18

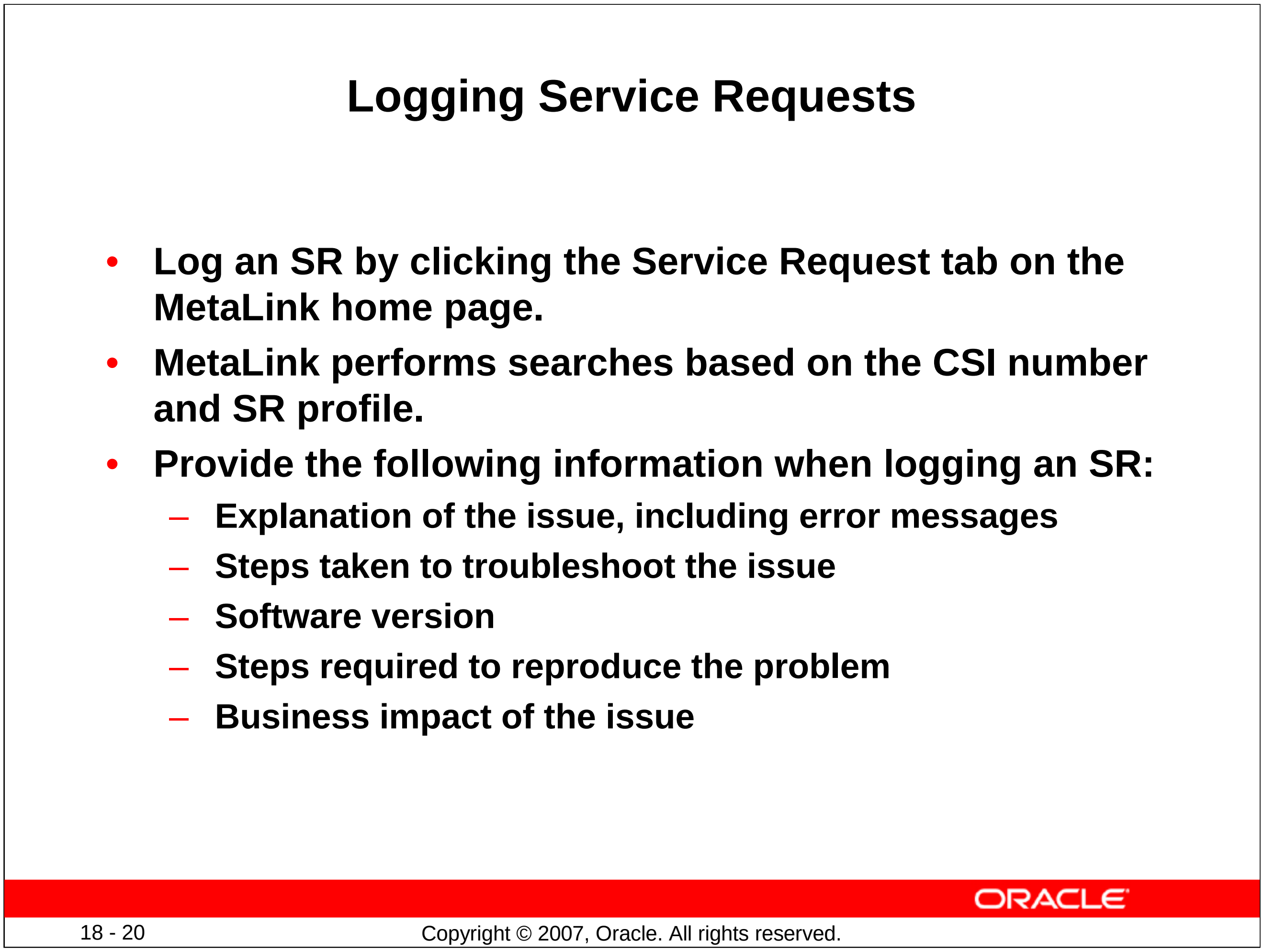
Researching an Issue

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

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

Researching an Issue (continued)

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



Logging Service Requests

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

18 - 20

Logging Service Requests

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

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

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

Logging Service Requests (continued)

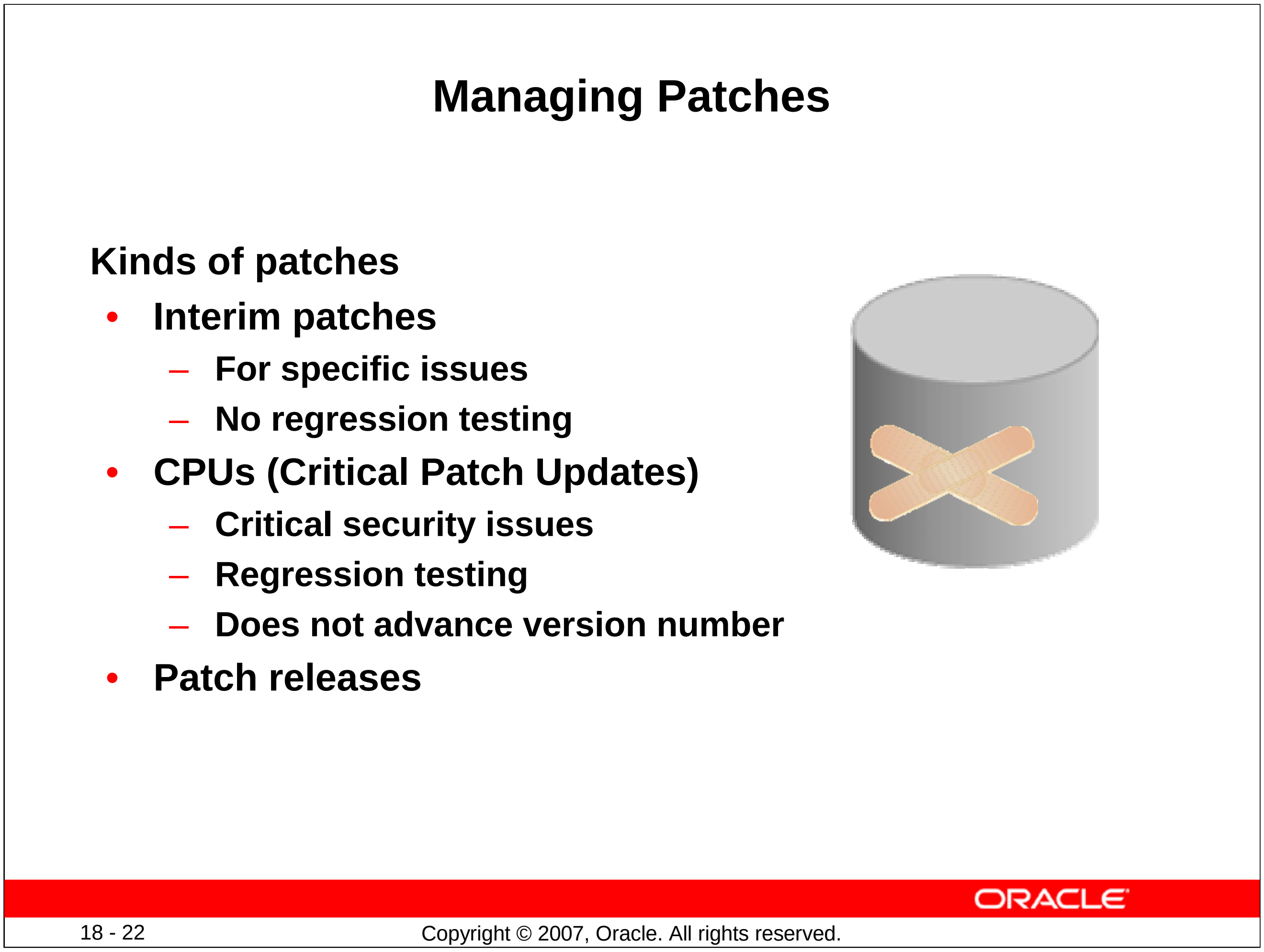
If the search results fail to resolve the issue, the SR creation process continues with a series of questions and requests for information. After the questions are answered, the SR is submitted electronically and routed to a support representative who analyzes the issue further. Any files, screenshots,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

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

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

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

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



Managing Patches

Kinds of patches

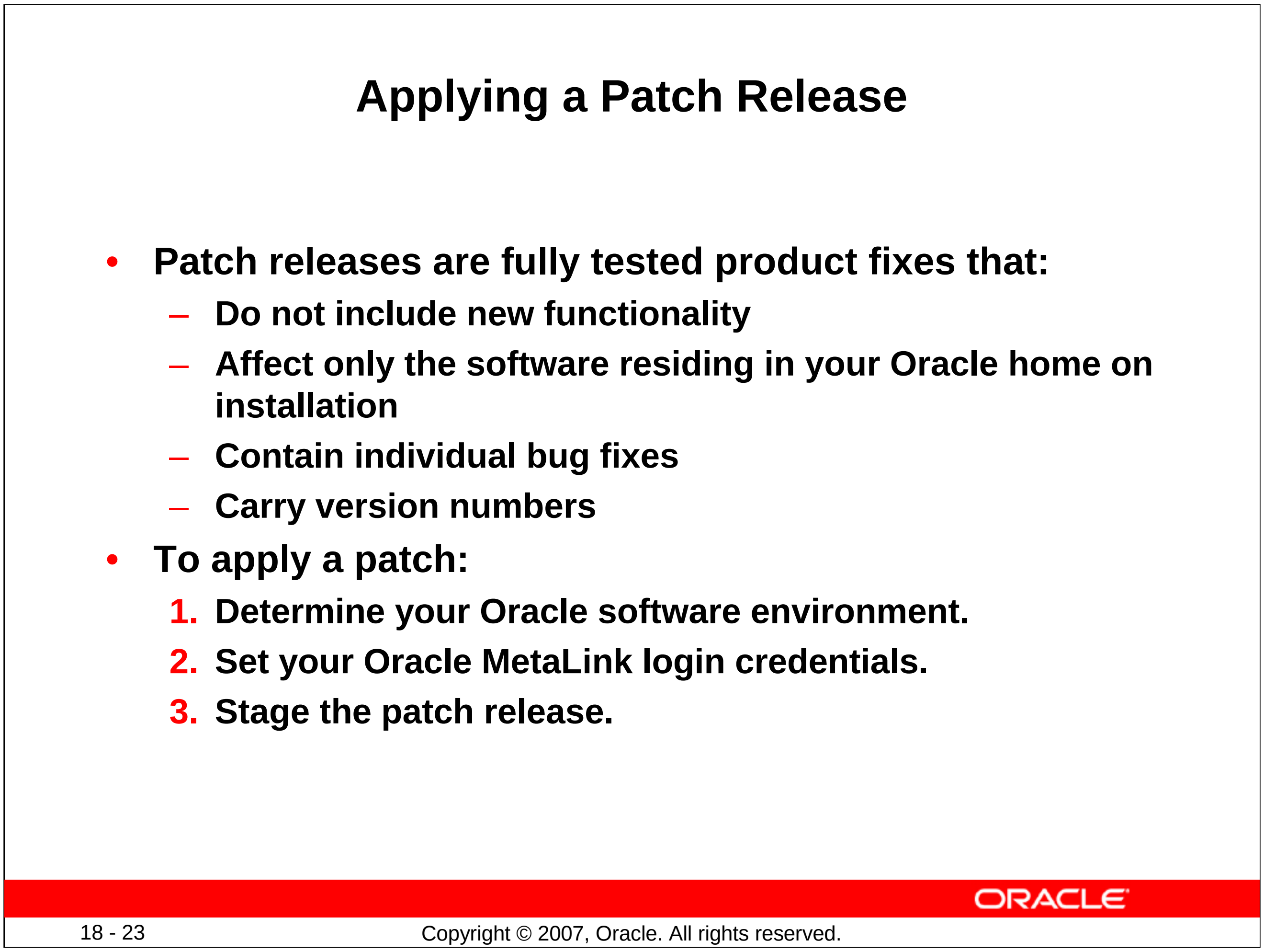
- Interim patches
  - For specific issues
  - No regression testing
- CPUs (Critical Patch Updates)
  - Critical security issues
  - Regression testing
  - Does not advance version number
- Patch releases

Managing Patches

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

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

  Oracle does extensive testing of Critical Patch Updates with our own applications, as well as running regression tests for the Critical Patch Updates themselves. To verify that a patch has been applied, query the inventory with `opatch -lsinventory` and see if the patch is listed.



Applying a Patch Release

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

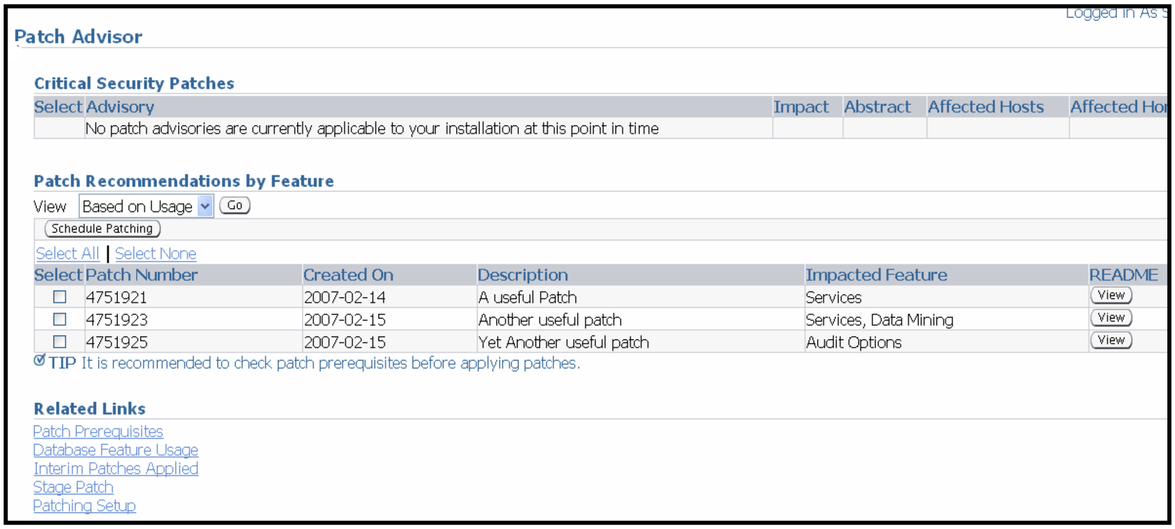
Applying a Patch Release

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

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

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

Using the Patch Advisor



Logged in As S

Patch Advisor

Critical Security Patches

Select	Advisory	Impact	Abstract	Affected Hosts	Affected Ho
	No patch advisories are currently applicable to your installation at this point in time				

Patch Recommendations by Feature

Select	Patch Number	Created On	Description	Impacted Feature	README
☐	4751921	2007-02-14	A useful Patch	Services	View
☐	4751923	2007-02-15	Another useful patch	Services, Data Mining	View
☐	4751925	2007-02-15	Yet Another useful patch	Audit Options	View

TIP It is recommended to check patch prerequisites before applying patches.

Related Links

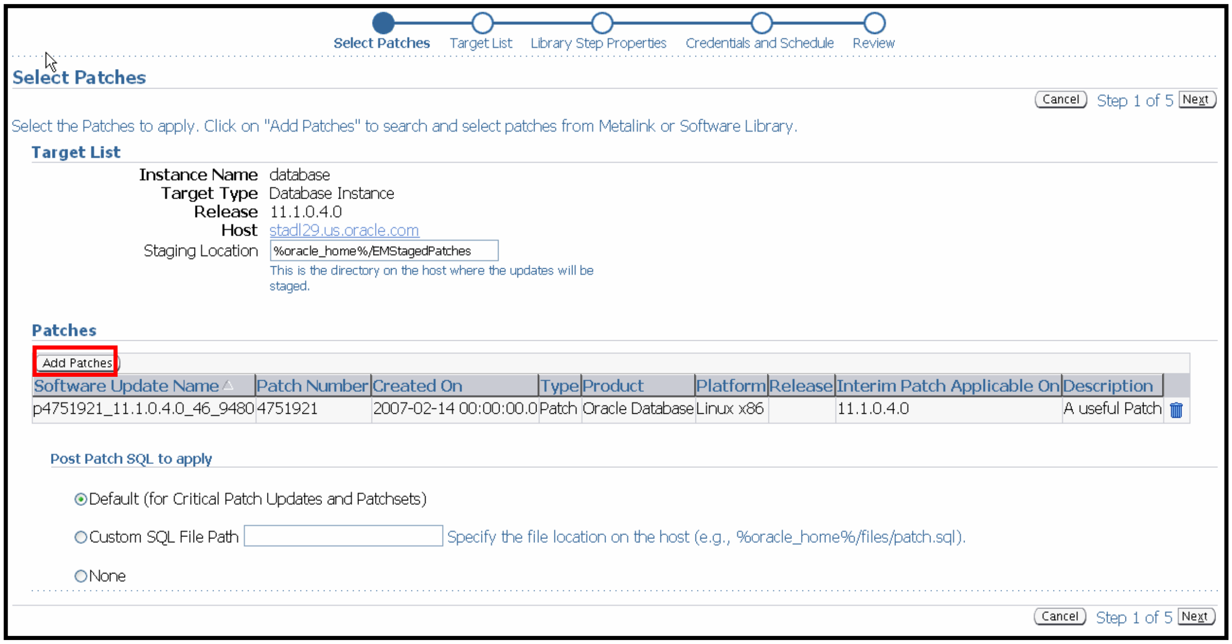
Patch Prerequisites
Database Feature Usage
Interim Patches Applied
Stage Patch
Patching Setup

Using the Patch Advisor

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

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

Using the Patch Wizard



Select Patches

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

Target List

Instance Name	database
Target Type	Database Instance
Release	11.1.0.4.0
Host	stadl29.us.oracle.com
Staging Location	%oracle_home%/EMStagedPatches This is the directory on the host where the updates will be staged.

Patches

Software Update Name	Patch Number	Created On	Type	Product	Platform	Release	Interim Patch Applicable On	Description
p4751921_11.1.0.4.0_46_9480	4751921	2007-02-14 00:00:00.0	Patch	Oracle Database	Linux x86		11.1.0.4.0	A useful Patch

Post Patch SQL to apply

● Default (for Critical Patch Updates and Patchsets)

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

○ None

Using the Patch Wizard

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

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

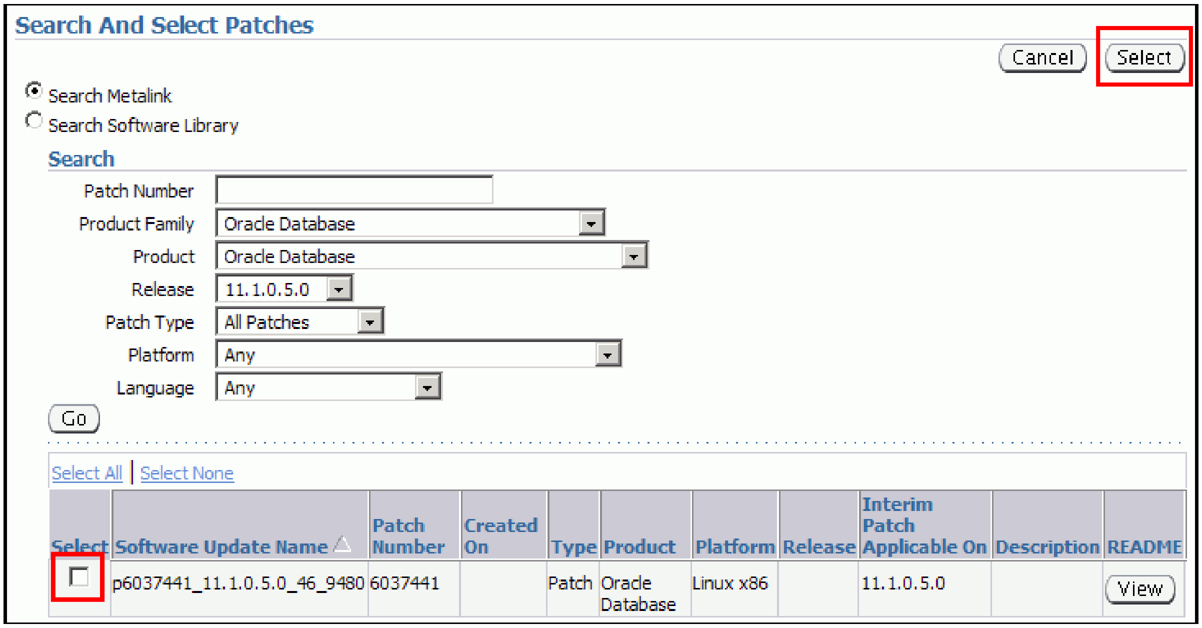
The Target List step is reserved for patching RAC and is skipped in Oracle Database 11*g* Release 1.

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

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

You then review the job and submit it.

Applying a Patch



Search And Select Patches

● Search Metalink
○ Search Software Library

Search

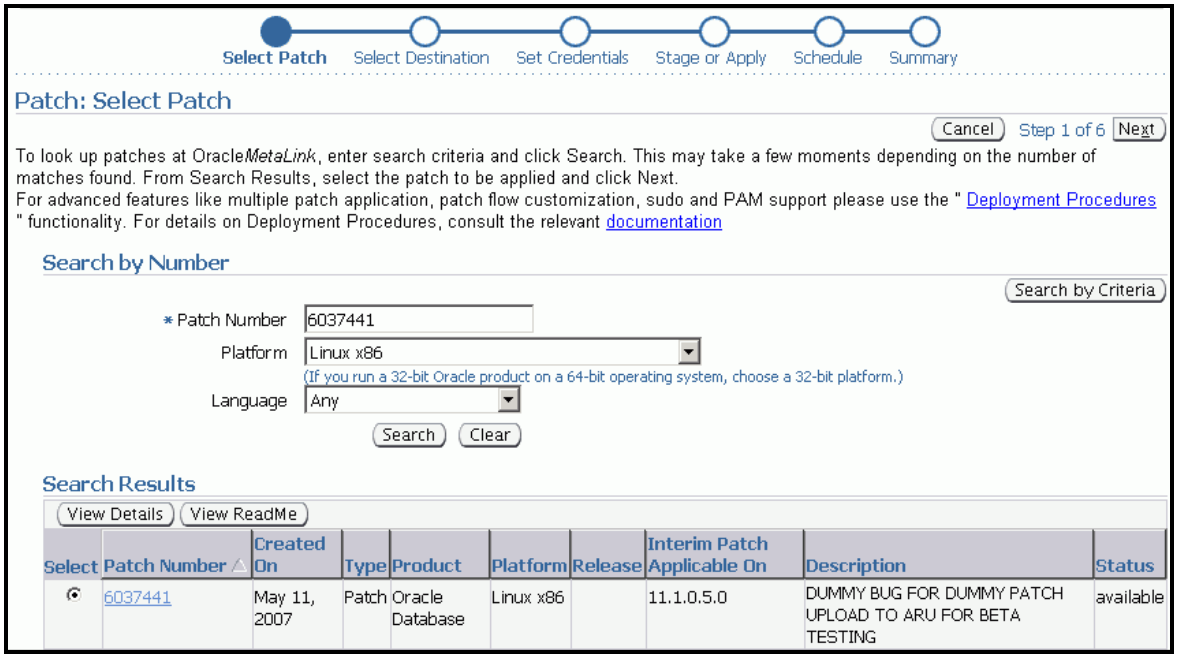
Patch Number	
Product Family	Oracle Database
Product	Oracle Database
Release	11.1.0.5.0
Patch Type	All Patches
Platform	Any
Language	Any

Select	Software Update Name	Patch Number	Created On	Type	Product	Platform	Release	Interim Patch Applicable On	Description	README
□	p6037441_11.1.0.5.0_46_9480	6037441		Patch	Oracle Database	Linux x86		11.1.0.5.0		View

Applying a Patch

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

Staging a Patch



Patch: Select Patch

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

Search by Number

* Patch Number	6037441
Platform	Linux x86 (If you run a 32-bit Oracle product on a 64-bit operating system, choose a 32-bit platform.)
Language	Any

Search Results

Select	Patch Number	Created On	Type	Product	Platform	Release	Interim Patch Applicable On	Description	Status
●	6037441	May 11, 2007	Patch	Oracle Database	Linux x86		11.1.0.5.0	DUMMY BUG FOR DUMMY PATCH UPLOAD TO ARU FOR BETA TESTING	available

Staging a Patch

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

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

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

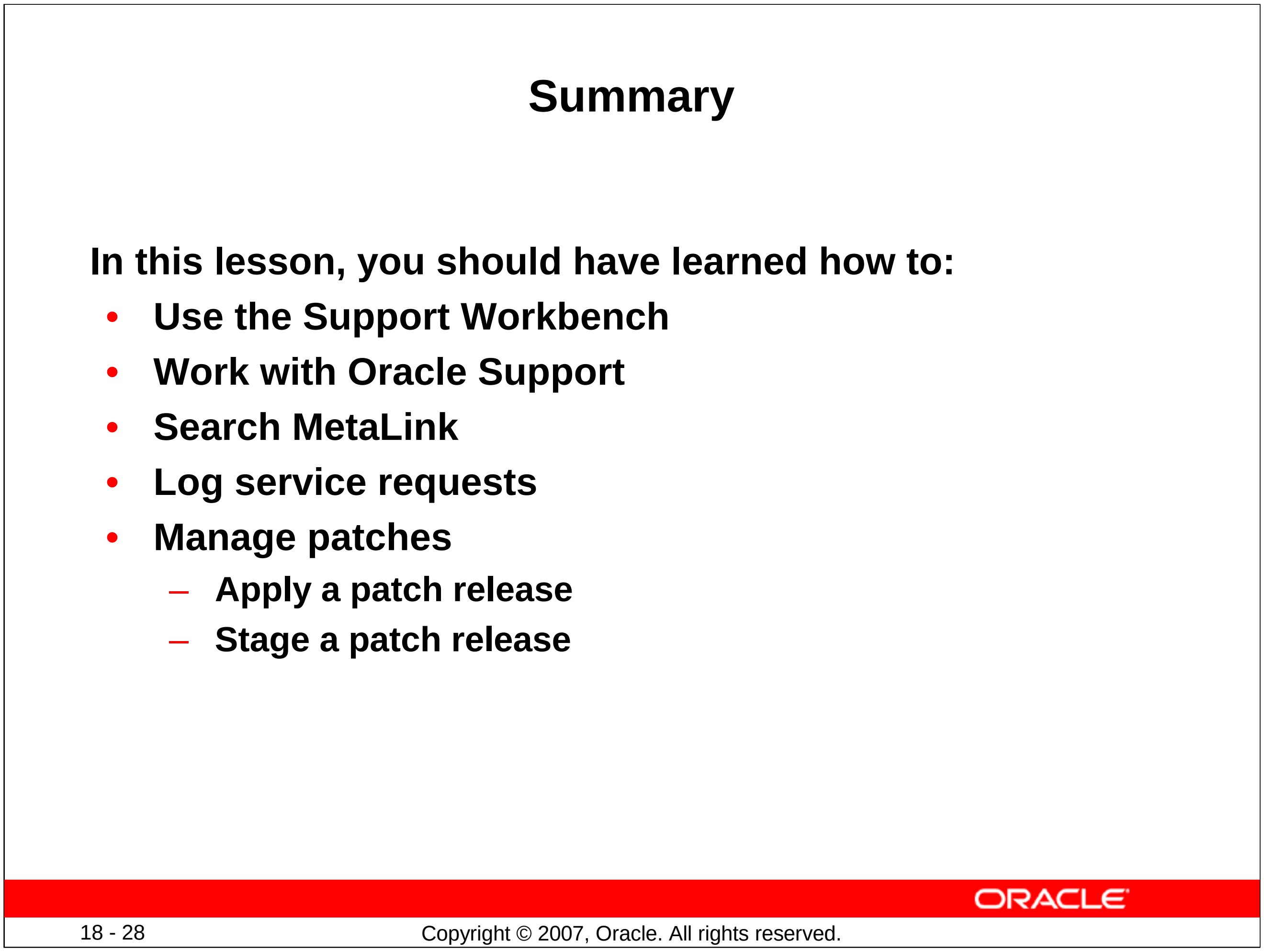
In the third step, provide the credentials of the OS user that is to do the patching, It is recommended that this be the user that owns the software installation.

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

The fifth step schedules the job.

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

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

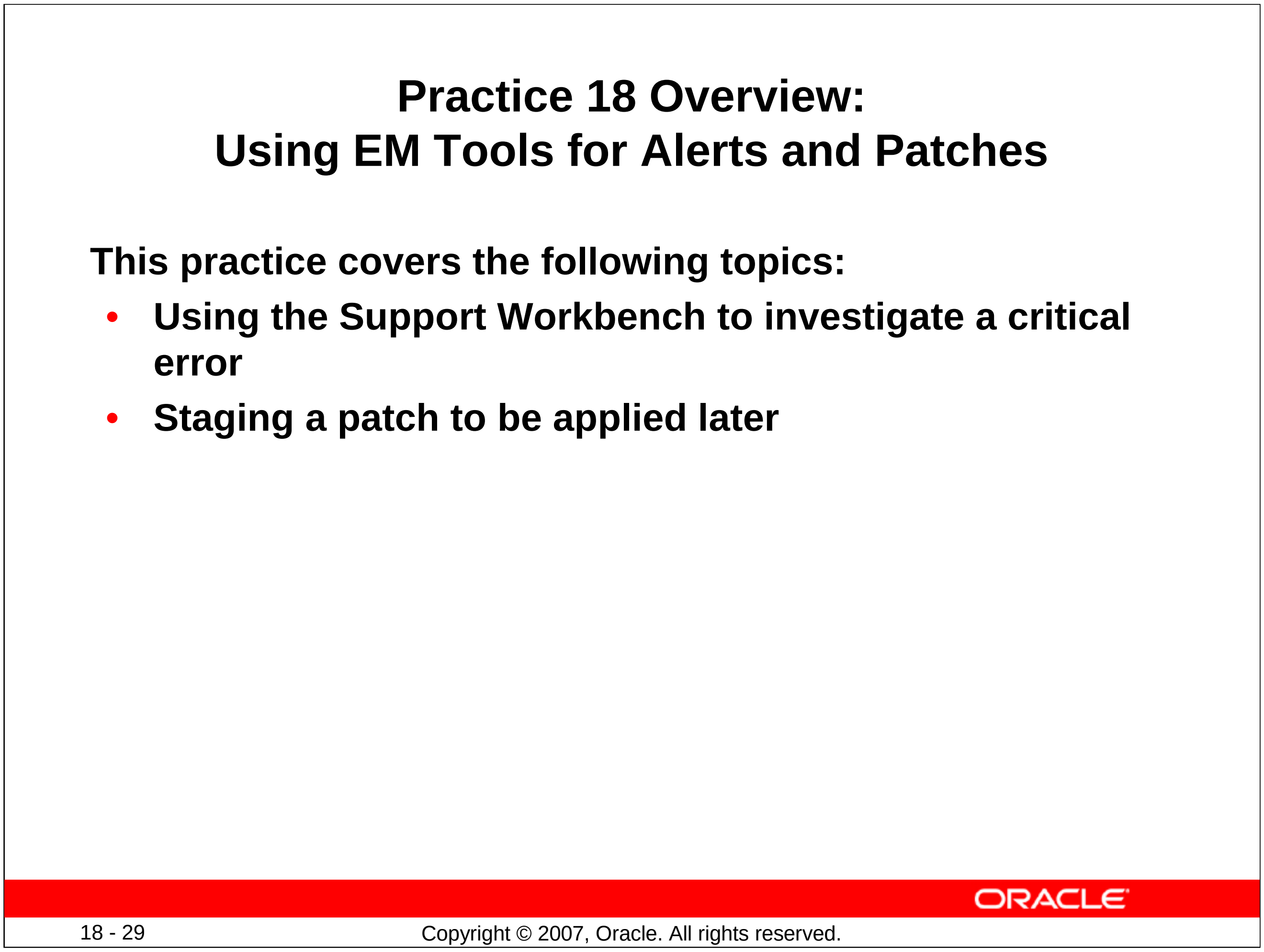


Summary

In this lesson, you should have learned how to:

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

18 - 28



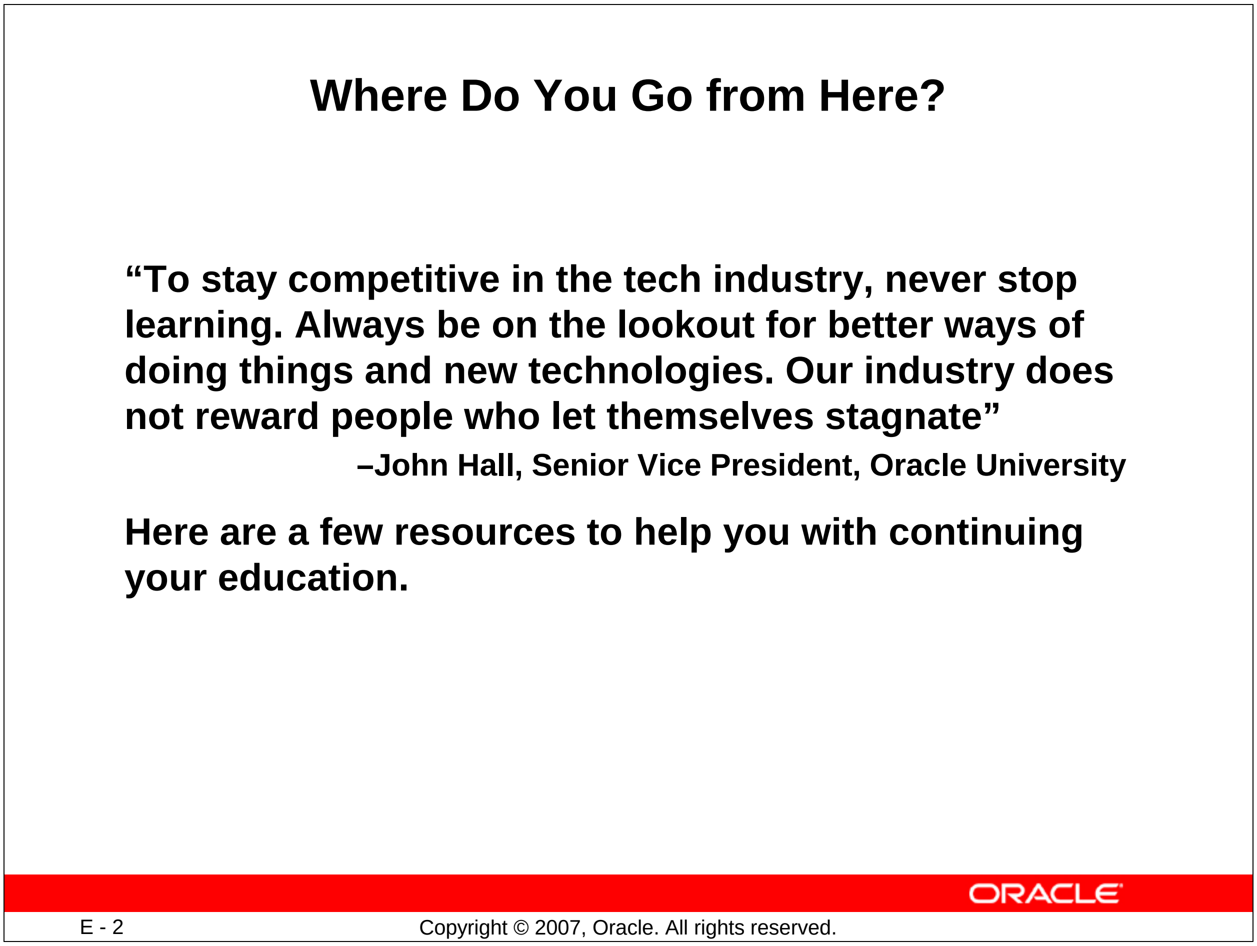
Practice 18 Overview: Using EM Tools for Alerts and Patches

This practice covers the following topics:

- Using the Support Workbench to investigate a critical error
- Staging a patch to be applied later

E

Next Steps: Continuing Your Education



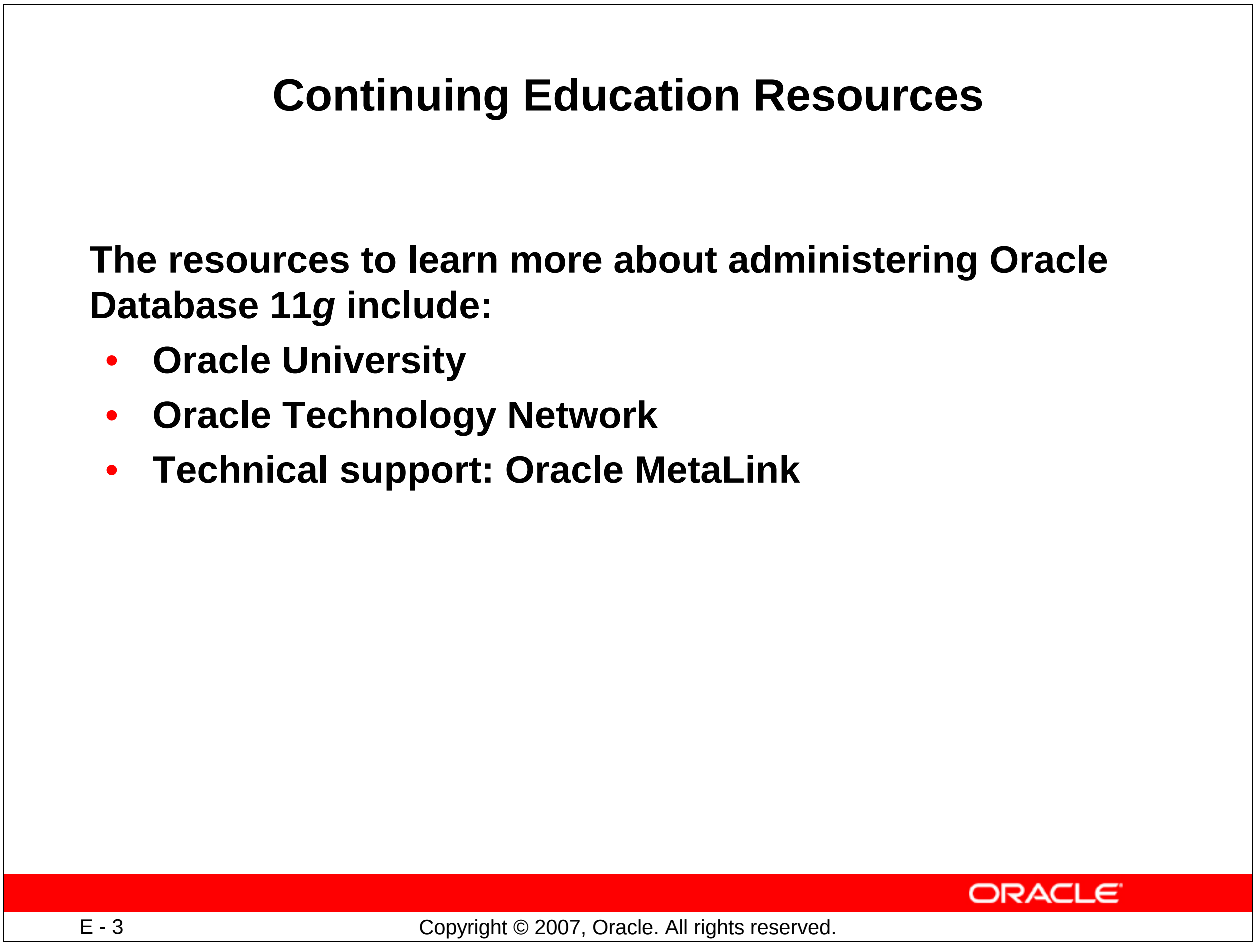
Where Do You Go from Here?

“To stay competitive in the tech industry, never stop learning. Always be on the lookout for better ways of doing things and new technologies. Our industry does not reward people who let themselves stagnate”

–John Hall, Senior Vice President, Oracle University

Here are a few resources to help you with continuing your education.

E - 2



Continuing Education Resources

The resources to learn more about administering Oracle Database 11*g* include:

- Oracle University
- Oracle Technology Network
- Technical support: Oracle MetaLink

Oracle University

http://education.oracle.com

ORACLE UNIVERSITY

UNITED STATES

ORACLE UNIVERSITY
100% Student Satisfaction
Course Schedule
Knowledge Center
Self-Study CD-ROM
User Adoption Services

PRODUCT COURSES
Database and Grids
Fusion Middleware
Development Tools
Collaboration
Data Warehouse
Linux | Java
E-Business Suite
PeopleSoft Enterprise
JD Edwards EnterpriseOne
JD Edwards World
Siebel
Retail Industry
Telecom Industry
Utilities Industry

Learn Oracle from Oracle! No one knows Oracle technology better than Oracle University.

- **Worldwide education services**
- **100% student satisfaction**
- **Learn with the format that best suits *your* needs:**
  - **Instructor-Led Inclass Training**
  - **Live Web Class**
  - **Self-Study CD-ROMs**
- **Certification**

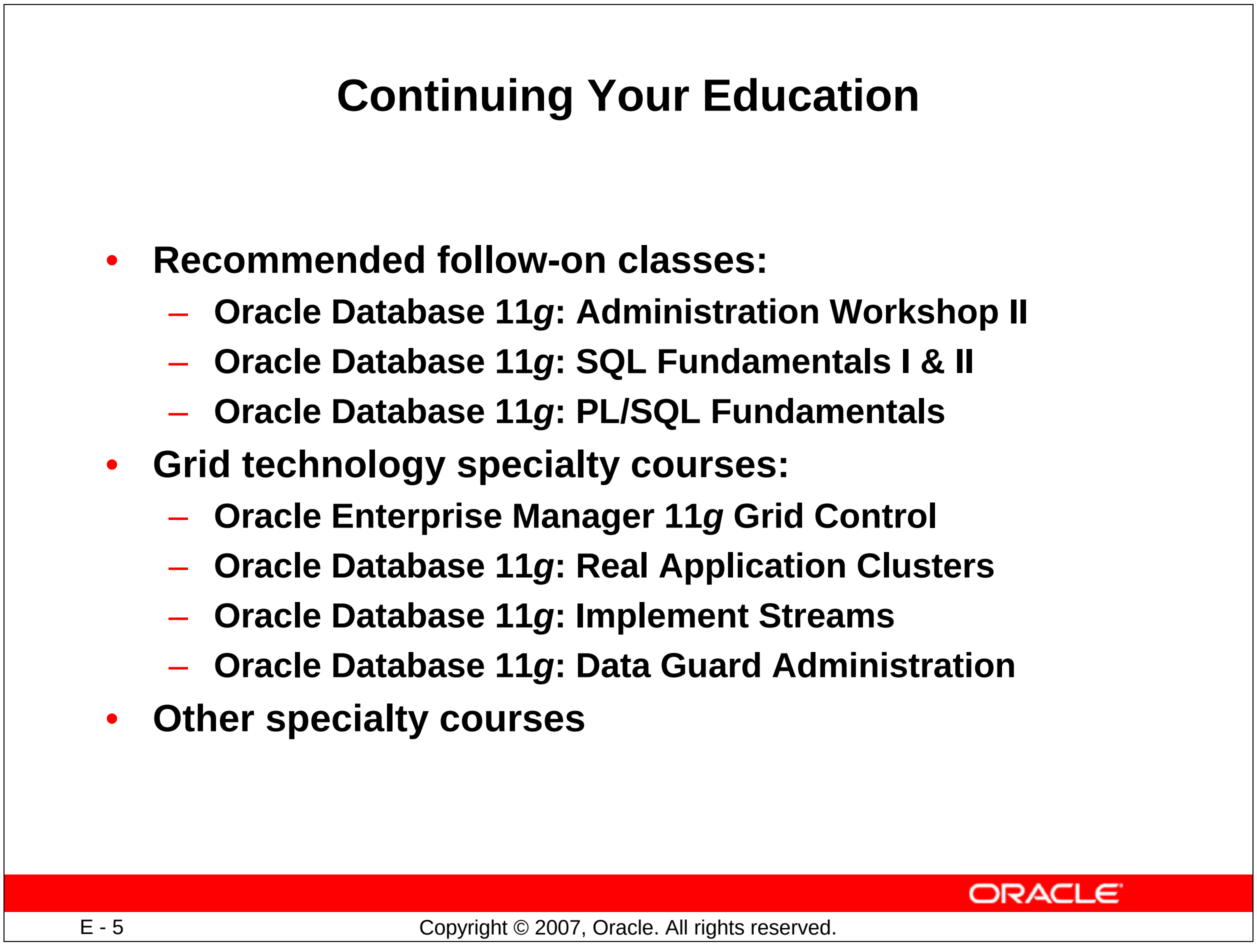
ORACLE

Oracle University

Oracle University is the world’s largest corporate educator with education centers around the globe. The goal is 100% student satisfaction.

Oracle certifications are tangible, industry-recognized credentials that provide measurable benefits to IT Professionals and their employers. Numerous certification paths exist, for example, for DBAs:

- Oracle Certified Associate (OCA)
- Oracle Certified Professional (OCP)
- Oracle Certified Master (OCM), and
- Specialty certifications, for example, Oracle 10*g*: Managing Oracle on Linux Certified Expert



Continuing Your Education

- **Recommended follow-on classes:**
  - **Oracle Database 11*g*: Administration Workshop II**
  - **Oracle Database 11*g*: SQL Fundamentals I & II**
  - **Oracle Database 11*g*: PL/SQL Fundamentals**
- **Grid technology specialty courses:**
  - **Oracle Enterprise Manager 11*g* Grid Control**
  - **Oracle Database 11*g*: Real Application Clusters**
  - **Oracle Database 11*g*: Implement Streams**
  - **Oracle Database 11*g*: Data Guard Administration**
- **Other specialty courses**

E - 5

Continuing Your Education

The *Oracle Database 11g: Administration Workshop II* course continues your training as a database administrator. You cover advanced database recovery strategies, performance monitoring and tuning, and distributed data concepts.

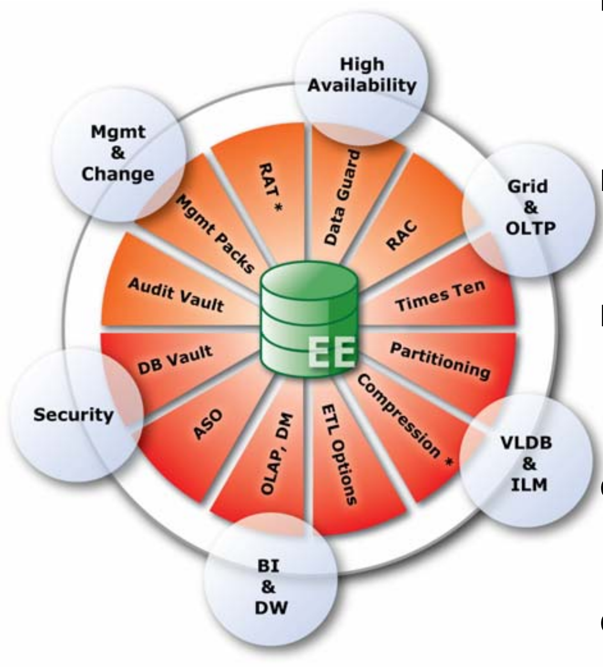
In this course, SQL and PL/SQL are discussed. Because both of these topics are vast, you are provided with only an overview. You can find additional training on these topics that can enhance your abilities as an administrator.

Oracle recommends that you complete the *Oracle Database 11g: Administration Workshop II* course, before beginning specialty courses.

Consult Oracle University’s website for an up-to-date list of all courses. Other specialty courses include:

- *Oracle Database 11g: Security*
- *Oracle Database 11g: Implement and Administer a Data Warehouse*

Database Specialty Areas



Modern Enterprise Grids

- Real Application Clusters
- Management packs
- TimesTen In-Memory Database

Information Lifecycle Management

- Partitioning
- Advanced Compression

Data Warehousing

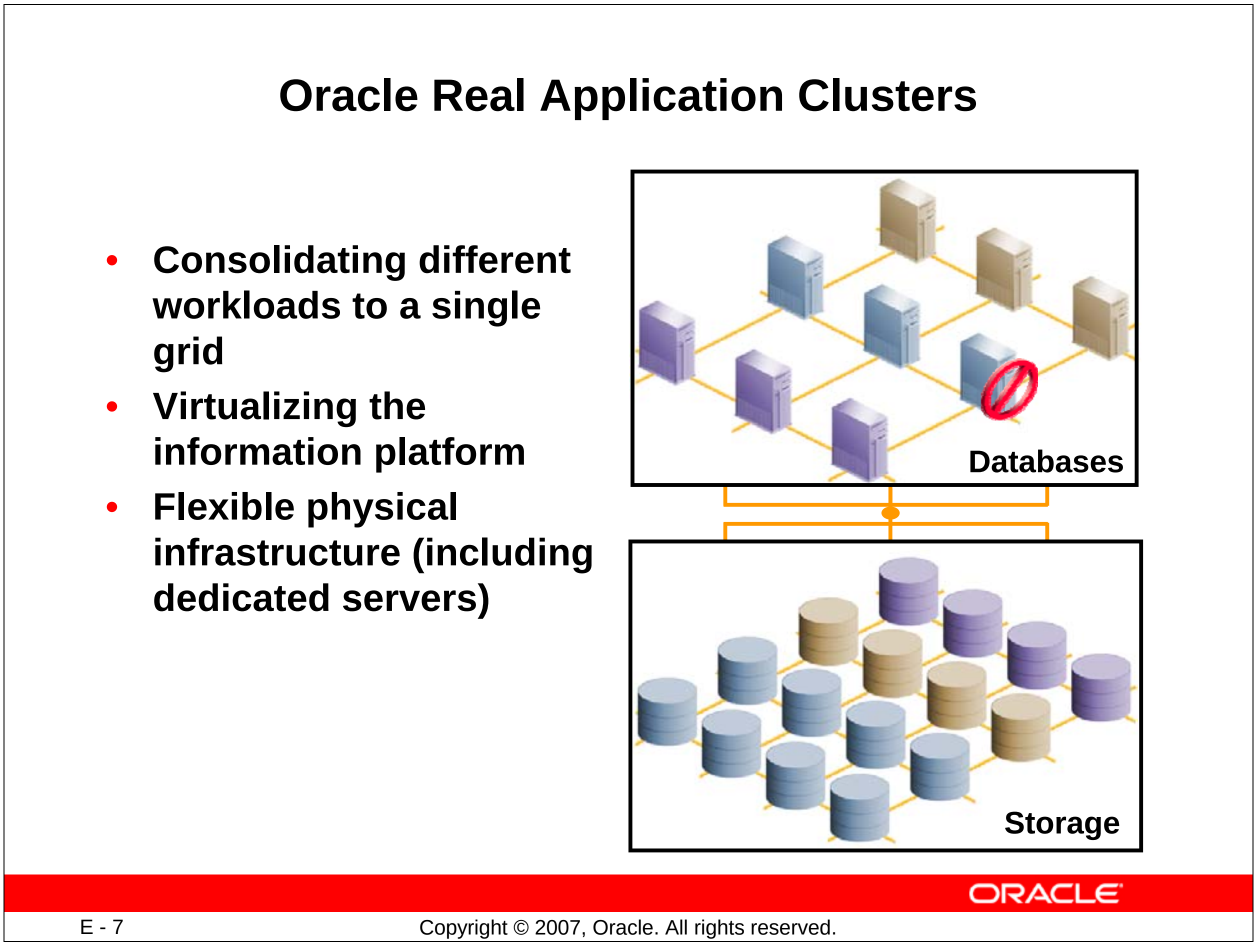
- Oracle Information Appliances
- OLAP, Mining, Warehouse Builder

Governance, Risk & Compliance

- Security Options
- Total Recall

Change management

- Real Application Testing



Oracle Real Application Clusters

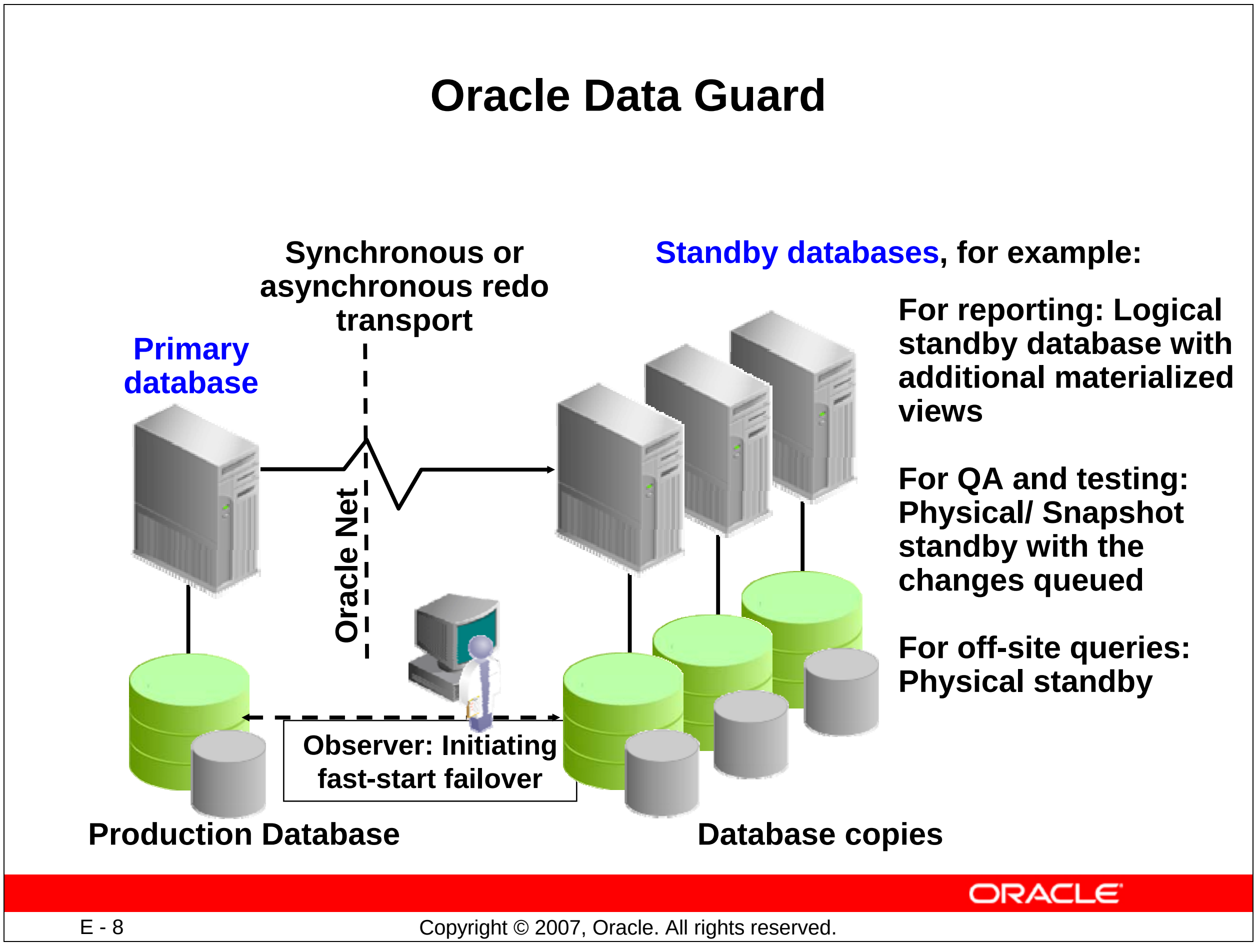
- Consolidating different workloads to a single grid
- Virtualizing the information platform
- Flexible physical infrastructure (including dedicated servers)

E - 7

Why use RAC?

Oracle Real Application Clusters (RAC) enables high utilization of a cluster of standard, low-cost modular servers such as blades. RAC offers automatic workload management for services. Services are groups or classifications of applications that comprise business components corresponding to application workloads. Services in RAC enable continuous, uninterrupted database operations and provide support for multiple services on multiple instances. You assign services to run on one or more instances, and alternate instances can serve as backup instances. If a primary instance fails, Oracle moves the services from the failed instance to a surviving alternate instance. Oracle also automatically load-balances connections across instances hosting a service.

RAC harnesses the power of multiple low-cost computers to serve as a single large computer for database processing, and provides the only viable alternative to large-scale SMP boxes for all types of applications. RAC, which is based on a shared-disk architecture, can grow and shrink on demand without the need to artificially partition data among the servers of your cluster. RAC also offers a single-button addition and removal of servers to a cluster. Thus, you can easily provide or remove a server to or from the database.



Oracle Data Guard

Primary database

Synchronous or asynchronous redo transport

Oracle Net

Standby databases, for example:

For reporting: Logical standby database with additional materialized views

For QA and testing: Physical/ Snapshot standby with the changes queued

For off-site queries: Physical standby

Observer: Initiating fast-start failover

Production Database

Database copies

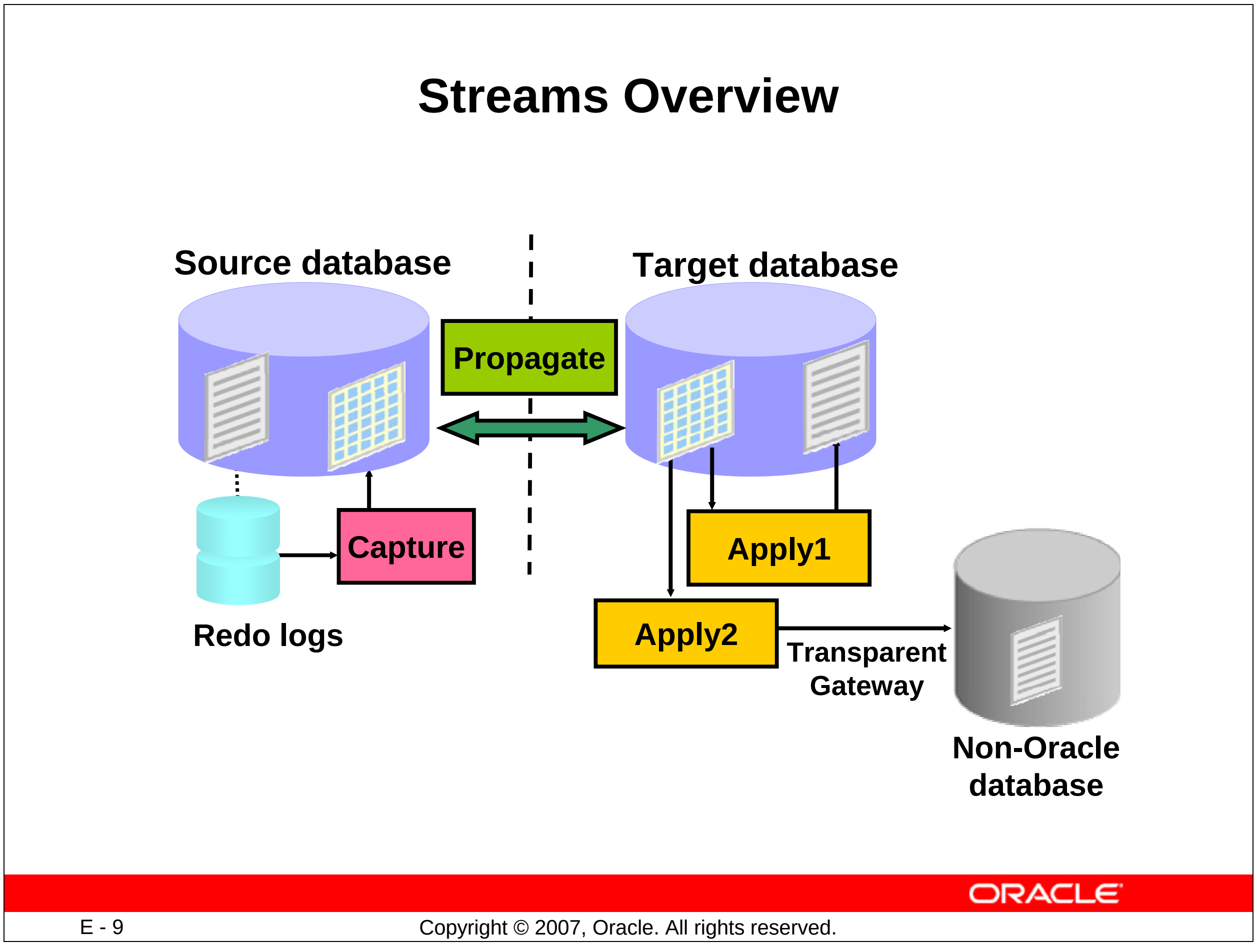
Oracle Data Guard

Oracle Data Guard is a management, monitoring, and automation software infrastructure that works with a production database and one or more standby databases to protect your data against failures, errors, and corruptions that might otherwise destroy your database. It protects critical data by providing facilities to automate the creation, management, and monitoring of the databases and other components in a Data Guard configuration. It automates the process of maintaining a copy of an Oracle production database (called a *standby database*) that can be used if the production database is taken offline for routine maintenance or becomes damaged.

In a Data Guard configuration, a production database is referred to as a *primary database*. A *standby database* is a synchronized copy of the primary database. Using a backup copy of the primary database, you can create from one to nine standby databases. The standby databases, together with the primary database, make up a Data Guard configuration. Each standby database is associated with only one primary database.

Note: You can use the Cascaded Redo Log Destinations feature to incorporate more than nine standby databases in your configuration.

Configuring standby redo log files is highly recommended on all standby databases in a Data Guard configuration, including the primary database to aid in role reversal.



Streams Overview

Streams Overview

A stream is a flow of information either within a database or from one database to another. Oracle Streams is a set of processes and database structures that enable you to share data and messages in a data stream. The unit of information that is put into a stream is called an event:

- DDL or DML changes, formatted as an LCR
- User-created events

Events are staged in and propagated between queues.

Most people think of Streams as replication where all databases can be updatable, and without platform or release considerations. Characteristics include:

- All sites: Active and updateable
- Automatic conflict detection and optional resolution
- Supporting data transformations
- Flexible configurations: n-way, hub & spoke, and so on
- Different database platforms, releases and schemas
- Providing high availability for applications (where update conflicts can be avoided or managed)

Oracle Streams: Basic Elements

By using Oracle Streams, you can share data and events in a data stream, either within a database or from one database to another.

Oracle Streams uses queues to stage events for propagation or consumption. You can use Oracle Streams to propagate events from one queue to another, and these queues can be in the same database or in different databases. You may stage two types of events in a queue used by Streams: captured events (logical change records, or LCRs) and user-enqueued events (which can be messages or LCRs):

- Changes to the database can be captured from the redo logs. You can then format these changes into LCRs. The LCRs can represent data manipulation language (DML) or data definition language (DDL) changes. The database where changes are generated in the redo log is called the source database.
- You can also enqueue user events explicitly with a user application. These explicitly enqueued events can be LCRs or user-created messages. A message is the smallest unit of information that is inserted into and retrieved from a queue. A message consists of data as well as information to govern the interpretation and use of the message data.

You can divide Oracle Streams into a small set of tasks. By configuring these tasks, you can control what information is put into a stream, how the stream flows from node to node, what happens to events in the stream as they flow into each node, and how the stream terminates.

You can customize each task to address specific requirements and business needs. The result is a new feature that provides greater functionality and flexibility than traditional solutions for capturing and managing events, and for sharing the events with other databases and applications. Oracle Streams provides the capabilities that are needed to build and operate distributed enterprises and applications, data warehouses, and high-availability solutions.

The three basic tasks of Oracle Streams are:

- **Capture:** To capture DML or DDL events automatically from the redo log. User-created events are not captured automatically but are placed into a queue via an explicit enqueue operation.
- **Staging:** To store and propagate events between databases. Propagation can be performed explicitly if needed.
- **Apply:** To apply DML or DDL events to a destination database or to pass the events to an application.

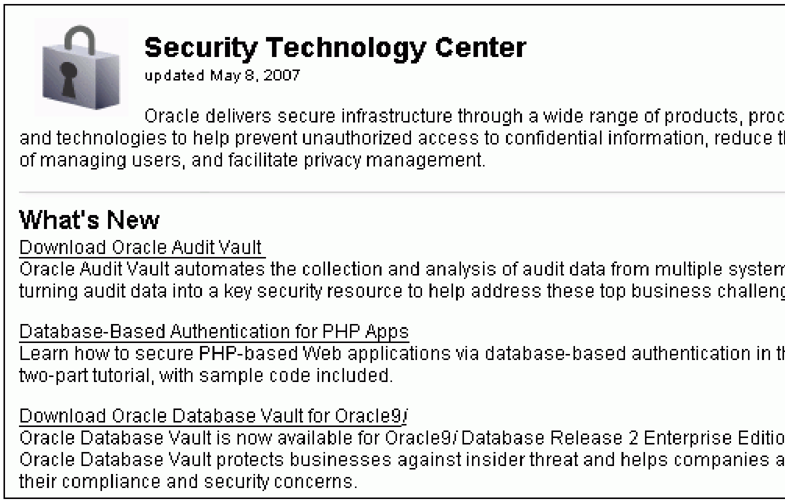
You can perform these tasks in a single database or combine them with tasks in other databases to form a distributed environment.

Multi-Database Streams

Events propagate between the staging areas in each database. The capture and consumption elements can be active in any database. For example, you can configure bidirectional data replication with a capture process, propagation job, and apply process at each site. Or, you can have a single-source system with capture and propagation at one site and apply at several other databases. You can also have an arbitrary number of databases. Some of the more complex environments may need hundreds of databases sharing information with Oracle Streams.

Security

http://www.oracle.com/technology/deploy/security/index.html



Security Technology Center

updated May 8, 2007

Oracle delivers secure infrastructure through a wide range of products, processes, and technologies to help prevent unauthorized access to confidential information, reduce the cost of managing users, and facilitate privacy management.

What's New

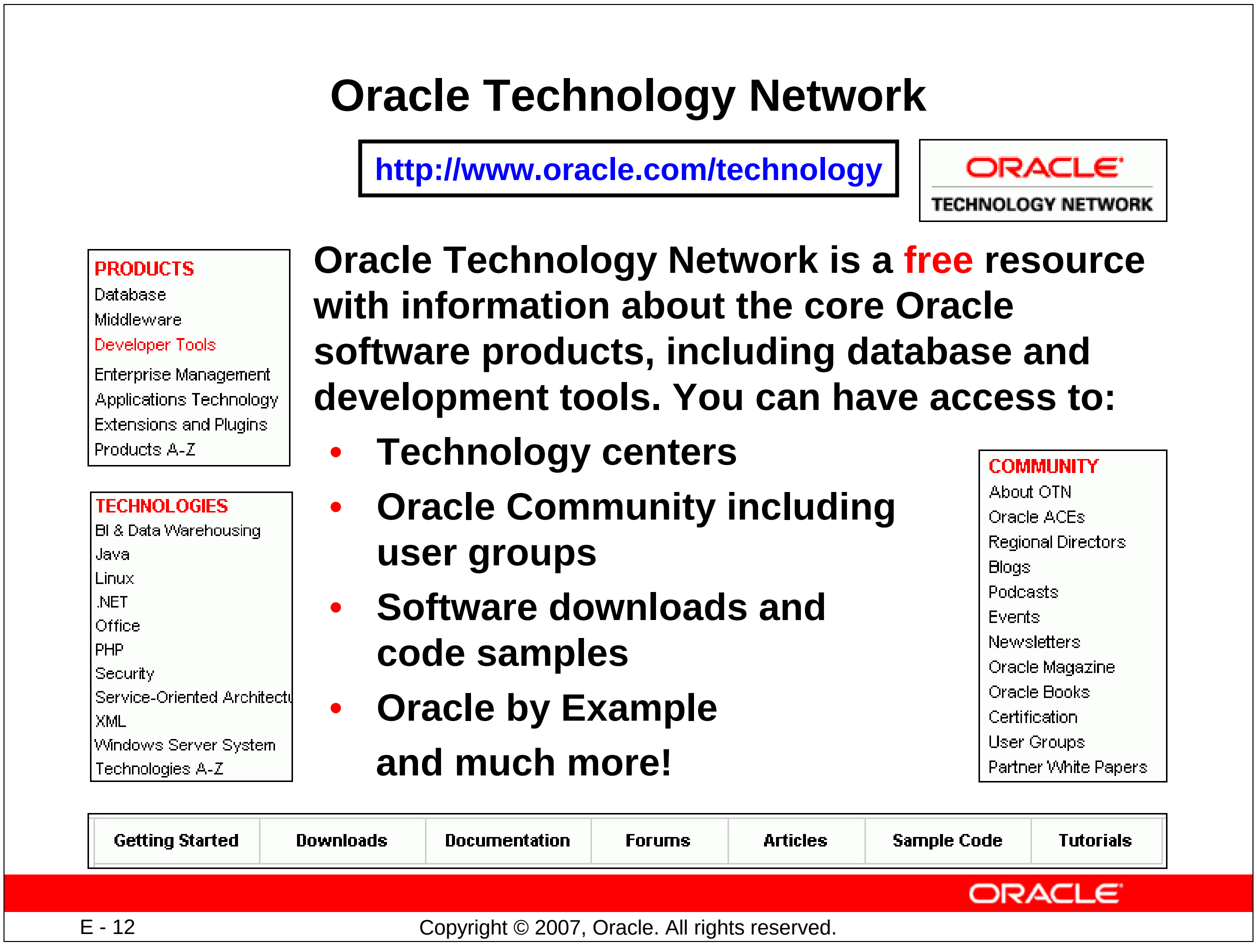
Download Oracle Audit Vault
Oracle Audit Vault automates the collection and analysis of audit data from multiple systems, turning audit data into a key security resource to help address these top business challenges.

Database-Based Authentication for PHP Apps
Learn how to secure PHP-based Web applications via database-based authentication in this two-part tutorial, with sample code included.

Download Oracle Database Vault for Oracle9*i*
Oracle Database Vault is now available for Oracle9*i* Database Release 2 Enterprise Edition. Oracle Database Vault protects businesses against insider threat and helps companies addre their compliance and security concerns.

Security

For more information about all security related aspects of the database, visit the “Security Technology Center” which is updated regularly.



Oracle Technology Network

http://www.oracle.com/technology

ORACLE
TECHNOLOGY NETWORK

PRODUCTS
Database
Middleware
Developer Tools
Enterprise Management
Applications Technology
Extensions and Plugins
Products A-Z

TECHNOLOGIES
BI & Data Warehousing
Java
Linux
.NET
Office
PHP
Security
Service-Oriented Architectu
XML
Windows Server System
Technologies A-Z

Oracle Technology Network is a free resource with information about the core Oracle software products, including database and development tools. You can have access to:

- **Technology centers**
- **Oracle Community including user groups**
- **Software downloads and code samples**
- **Oracle by Example**

and much more!

COMMUNITY
About OTN
Oracle ACEs
Regional Directors
Blogs
Podcasts
Events
Newsletters
Oracle Magazine
Oracle Books
Certification
User Groups
Partner White Papers

Getting Started | Downloads | Documentation | Forums | Articles | Sample Code | Tutorials

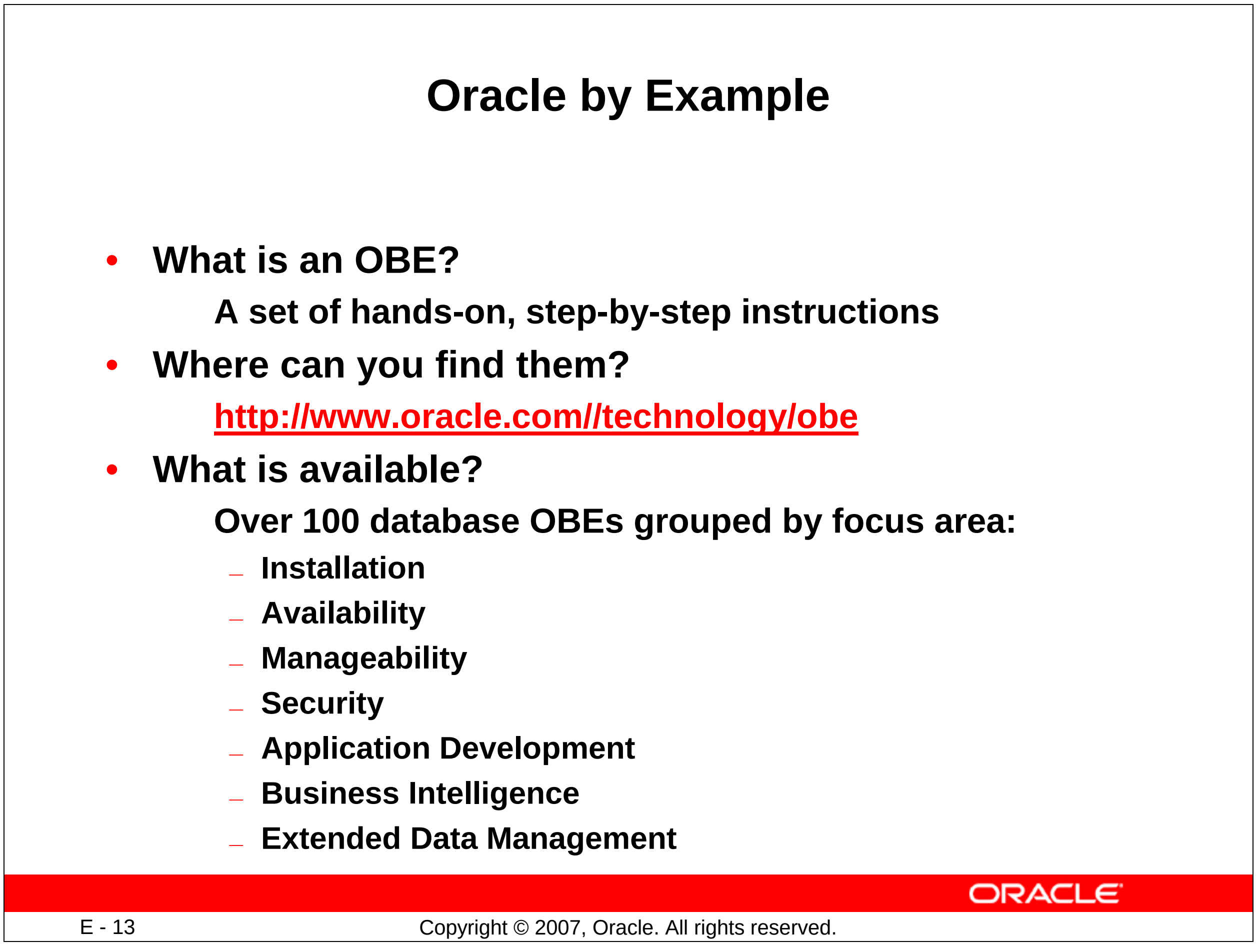
ORACLE

E - 12

Oracle Technology Network

Oracle Technology Network (OTN) hosts the latest news about Oracle technology and products. Additionally, OTN provides peer-to-peer forums, white papers, security bulletins, and other vital information for the Oracle professional.

In addition to tips, tricks, and techniques for getting the most out of your Oracle software, you can download that software from OTN. Remember: All software downloads are free, and each comes with a development license that allows you to use full versions of the products only when developing and making prototypes your applications.



Oracle by Example

- **What is an OBE?**

  A set of hands-on, step-by-step instructions

- **Where can you find them?**

  http://www.oracle.com//technology/obe

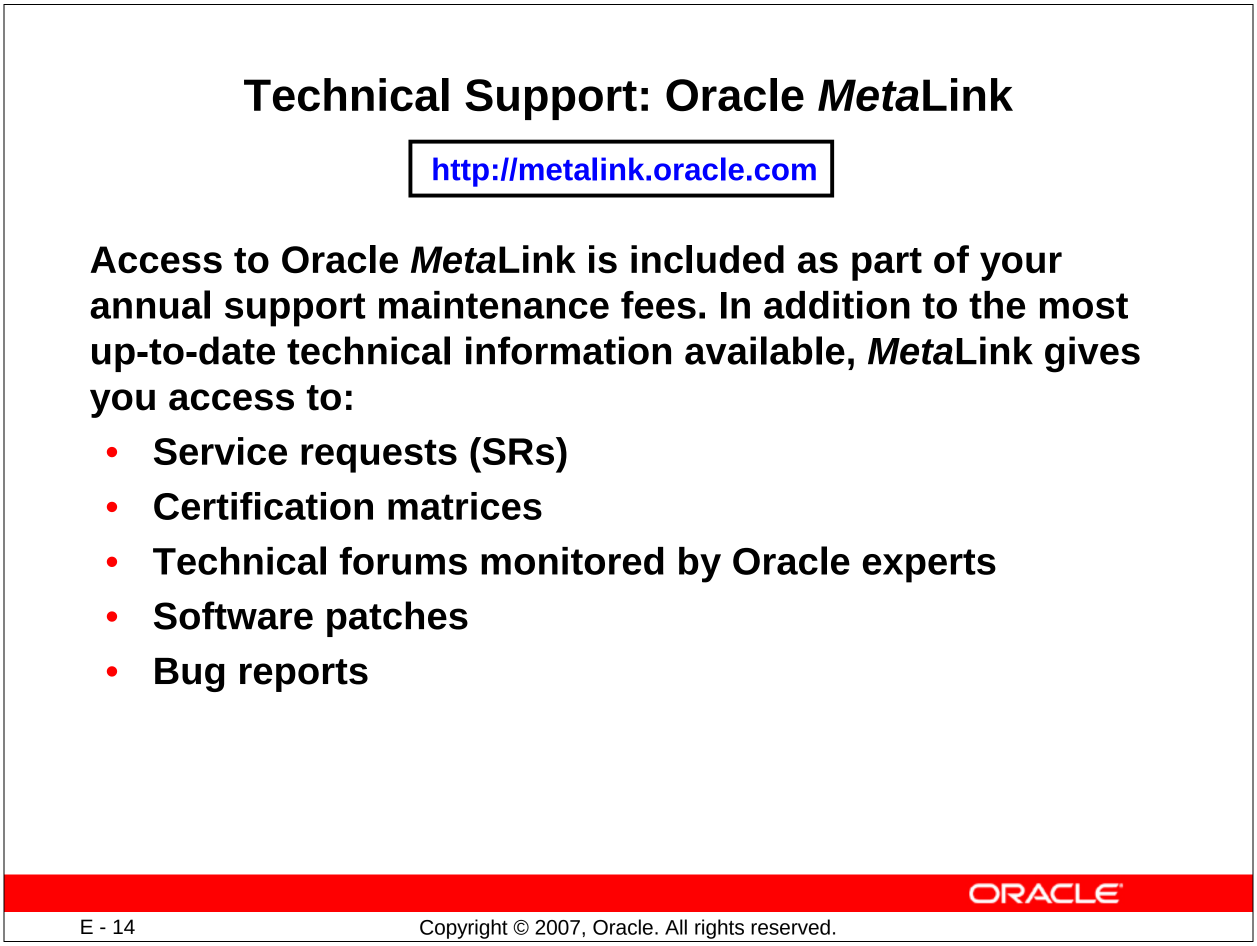
- **What is available?**

  Over 100 database OBEs grouped by focus area:

  - **Installation**
  - **Availability**
  - **Manageability**
  - **Security**
  - **Application Development**
  - **Business Intelligence**
  - **Extended Data Management**

Oracle by Example

The Oracle by Example (OBE) series provides hands-on, step-by-step instructions on how to use various new features of Oracle products. OBEs help to reduce the time spent on learning a new product capability and enhance the users' understanding of how the feature can be implemented in their environment. Currently, OBEs are available for the Oracle database, Oracle Application Server, and Oracle Collaboration Suite. OBEs can be accessed at http://www.oracle.com/technology/obe.



Technical Support: Oracle *Meta*Link

http://metalink.oracle.com

Access to Oracle *Meta*Link is included as part of your annual support maintenance fees. In addition to the most up-to-date technical information available, *Meta*Link gives you access to:

- **Service requests (SRs)**
- **Certification matrices**
- **Technical forums monitored by Oracle experts**
- **Software patches**
- **Bug reports**

E - 14

Oracle *Meta*Link

Oracle *Meta*Link is your gateway to Oracle’s Support resources. Here, you find answers to the most common issues facing Oracle administrators and developers, as well as resources to solve many of those issues.

Like Oracle Technology Network, *Meta*Link includes the most recent headlines about issues that affect the Oracle professional.

Thank You!

We hope your experience with Oracle University has been enjoyable. We welcome your feedback on how we can improve to better meet your needs:

- End-of-course evaluations
- Oracle University Office of Customer Satisfaction
- Oracle Education Services

We hope to see you in class again soon.

E - 15

Thank You!

Oracle University’s mission is to enhance the adoption of Oracle technology. Our goal is to partner with you, providing information that is pertinent, timely, and relevant to your needs.

Please take a minute to complete the end-of-course evaluation and let us know how we can serve you better. In the U.S., feel free to e-mail our office of customer satisfaction at:

`customersat_us@oracle.com`

If you have questions about continuing your Oracle education, need help finding a class, or want to arrange for on-site training at your company, contact Oracle Education Services for assistance. In the U.S., dial 800.529.0165. For contact numbers outside the U.S., visit the following Web site:

http://www.oracle.com/education/index.html?contact.html

Thanks again and hope to see you in another class!