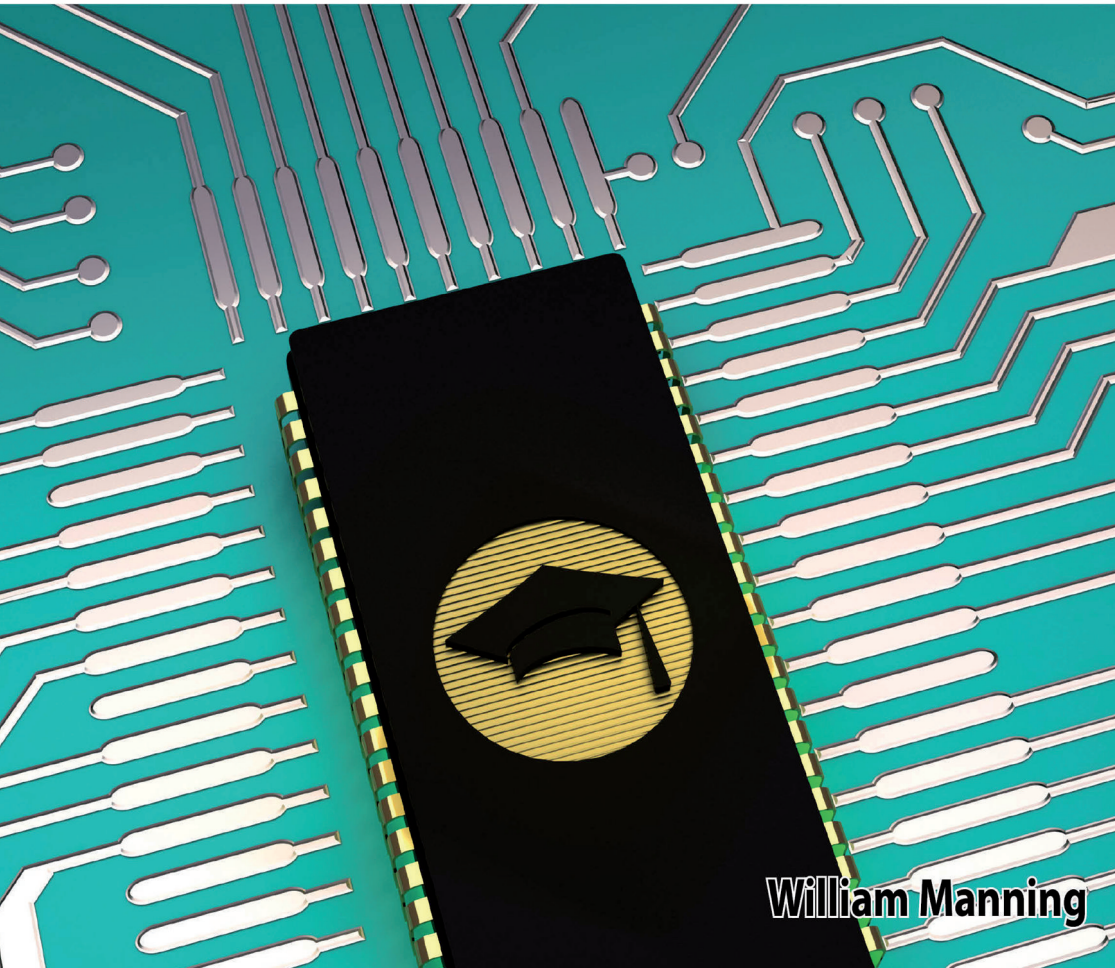
Self Study Exam OCP Preparation Guide - Covers all OCP Exam Topics

OCP Oracle Database 10g Database Administrator Certification

Exam Preparation Course in a Book for Passing the

OCP

The How To Pass on Your First Try Certification Study Guide



William Manning

Foreword

This Exam Preparation book is intended for those preparing for the Oracle Database 10g Database Administrator OCP certification.

This book is **not** a replacement for completing a course. This is a study aid to assist those who have completed an accredited course and preparing for the exam.

Do not underestimate the value of your own notes and study aids. The more you have, the more prepared you will be.

Due to licensing rights, we are unable to provide actual Oracle Database 10g Database Administrator OCP Exam. However, the study notes and sample exam questions in this book will allow you to more easily prepare for an Oracle Database 10g Database Administrator OCP exam.

Good luck!

Ivanka Menken
Executive Director
The Art of Service

Table of Contents

FOREWORD **1**

TABLE OF CONTENTS **2**

1 ORACLE DATABASE 10G ADMINISTRATION II CERTIFICATION **9**

2 EXAM SPECIFICS **10**

3 ORACLE DATABASE 10G ARCHITECTURE **11**

3.1 ORACLE 10G 11
3.2 DATABASE CONCEPTS 12
3.2.1 Structured Query Language *12*
3.2.2 Database Constructs *13*
3.2.3 Metadata Views *13*
3.2.4 Relationships and Constraints *14*
3.2.5 Segment Types *15*
3.2.6 Communicating Between Segments *16*
3.3 ORACLE ARCHITECTURE 17
3.3.1 Database Administrator *17*
3.3.2 Database Server Architecture *17*
3.3.3 System Global Area *18*
3.3.4 Basic Initialization Parameters *19*
3.3.5 Oracle Background Processes *21*
3.3.6 Oracle Database *22*
3.4 INSTALLING ORACLE 10G 24
3.4.1 Review Documentation *24*
3.4.2 System Requirements *25*
3.4.3 Installation Plan *25*
3.4.4 Oracle Universal Installer *26*

4 DATABASE CREATION **28**

4.1 ORACLE DATABASE 28
4.1.1 Management Framework *28*
4.1.2 Oracle Management Agents *29*
4.1.3 Oracle Database Configuration Assistant (ODCA) *30*
4.1.4 Database Templates *30*
4.1.5 Database Credentials *31*
4.1.6 Storage Options *31*
4.1.7 Recovery Configuration *33*
4.1.8 Database Content *34*
4.1.9 Initialization Parameters *34*
4.1.10 Database Storage *36*
4.1.11 Creation Options *36*

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

4.2 OTHER DBCA ACTIVITIES 37
4.2.1 Configuring the Database *37*
4.2.2 Deleting the Database *37*
4.2.3 Managing Templates *37*
4.3 PARAMETER INITIALIZATION FILES 38
4.4 START AND STOP DATABASES 39
4.4.1 Administrative Authorization *39*
4.4.2 Startup Modes *40*
4.4.3 Special Startup Modes *41*
4.4.4 EM Database Control Startup *41*
*4.4.5 SQL*Plus Startup* *42*
4.4.6 Database Shutdown Processes *42*
4.4.7 EM Database Control Shutdown *43*
*4.4.8 SQL*Plus Shutdown* *44*
4.4.9 Alert Log Monitoring *44*

5 DATABASE STORAGE 45

5.1 STORAGE STRUCTURES 45
5.1.1 Tablespace Management *45*
5.1.2 Creating Tablespaces *46*
5.1.3 Other Tablespaces *48*
5.1.4 Removing and Modifying Tablespace *49*
5.1.5 Tablespace Information *50*
5.2 DATAFILE MANAGEMENT 51
5.3 SCHEMA OBJECTS 52
5.3.1 Datatypes *53*
5.3.2 Tables *56*
5.3.3 Creating Tables *57*
5.3.4 Table Management *58*
5.3.5 Constraints *59*
5.3.6 Indexes *61*
5.3.7 Managing Indexes *61*
5.3.8 Views *62*
5.3.9 Sequences *63*

6 RECOVERY MANAGER 64

6.1 RECOVERY MANAGER FEATURES 64
6.1.1 RMAN Components *65*
6.1.2 RMAN Usage *66*
6.1.3 Repository and Control Files *67*
6.1.4 Recovery Catalog *67*
6.2 STARTING AND CONNECTING TO RMAN 69
6.2.1 Connecting to Target Database *69*

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

6.2.2 Connecting to Recovery Catalog *70*
6.3 CONFIGURING RMAN 70
6.3.1 Media Management Layer *70*
6.3.2 Channel Allocation *70*
6.3.3 RMAN Parameters and Persistent Settings *72*
6.3.4 EM Configuration of RMAN Settings *72*
6.3.5 Retention Policies *73*
6.3.6 Control File Autobackup *73*
6.4 USING RECOVERY MANAGER 74
6.4.1 RMAN Commands *74*
6.4.2 Backup Sets and Image Copies *77*
6.4.3 Compressed Backups *78*
6.4.4 Full and Incremental Backups *78*
6.4.5 Parallelization *79*
6.4.6 Backup Options *79*
6.4.7 Block Change Tracking *80*
6.4.8 Monitoring Backups *81*
6.4.9 LIST Commands *81*
6.4.10 REPORT Commands *82*

7 DATABASE RECOVERY 83

7.1 RECOVERY PROCESS 83
7.2 SERVER-MANAGED RECOVERY 83
7.2.1 User-Managed Recovery *84*
7.2.2 Recovering Control Files *84*
7.2.3 Performing Incomplete Recoveries *85*
7.2.4 Recovery Through Enterprise Manager *86*
7.2.5 RESETLOGS Operation *86*
7.3 USING THE FLASHBACK DATABASE 87
7.3.1 Flashback Databases *87*
7.3.2 Managing the Flash Recovery Area *88*
7.3.3 Configuring the Flash Memory Area *89*
7.3.4 Using the Flash Memory Area *89*
7.3.5 Backup the Flash Recovery Area *91*
7.3.6 Configuring the Flashback Database *91*
7.3.7 Using RMAN with the Flashback Database *91*
7.3.8 Monitoring the Flashback Database *92*
7.3.9 Using EM with the Flashback Database *92*
7.4 NON-CRITICAL LOSSES 93
7.4.1 Temporary Tablespaces *93*
7.4.2 Redo Log Files *94*
7.4.3 Index Tablespaces *94*
7.4.4 Indexes *95*

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

7.4.5 Read-Only Tablespaces *95*
7.4.6 Password File *95*
7.5 USER ERROR RECOVERY 96
7.5.1 Overview of Technologies *96*
7.5.2 Using Flashdrop Drop *96*
7.5.3 Flashback Dropped Tables *98*
7.5.4 Flashback Versions Query *98*
7.5.5 Flashback Transaction Query *99*
7.5.6 Flashback Table *100*
7.6 BLOCK CORRUPTION 100
7.6.1 Symptoms of Block Corruption *100*
7.6.2 Detecting and Resolve Corruption *101*
7.6.3 Block Media Recovery *103*

8 AUTOMATED SERVICES 104

8.1 AUTOMATIC DATABASE MANAGEMENT 104
8.1.1 Automatic Workload Repository (AWR) *105*
8.1.2 Active Session History *108*
8.1.3 Snapshots *109*
8.1.4 AWR Baselines *110*
8.1.5 AWR Views *110*
8.1.6 AWR Reports *110*
8.2 SERVER GENERATED ALERTS 111
8.2.1 Alert Process *111*
8.2.2 Configuring Thresholds *111*
8.3 AUTOMATIC ROUTINE ADMINISTRATION TASKS 113
8.3.1 Automatic Statistics Collection *114*
8.3.2 Dictionary Statistics *115*
8.3.3 System Statistics *115*
8.3.4 Operating System Statistics *115*
8.3.5 User Defined Statistics *116*
8.3.6 Collecting Statistics *116*
8.3.7 Volatile Objects *117*
8.3.8 DML Table Changes *117*
8.3.9 STATISTICS_LEVEL Parameter *118*
8.4 ADVISORY FRAMEWORK 119
8.4.1 Advisors *119*
8.4.2 Invoking Advisors *120*
8.4.3 DBMS_ADVISOR Package *121*
8.4.4 Advisor Views *122*
8.4.5 Automatic Database Diagnostic Monitor *122*
8.4.6 SQL Tuning Advisor *123*
8.5 AUTOMATIC STORAGE MANAGEMENT 125

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

8.5.1 ASM Architecture *125*
8.5.2 ASM Instance *126*
8.5.3 Dynamic Performance Views *127*
8.5.4 File Types and Templates *128*
8.5.5 ASM Disk Groups *129*
8.5.6 Recovery with ASM *131*
8.6 GLOBALIZATION SUPPORT 132
8.6.1 Features for Globalization *132*
8.6.2 Architecture for Globalization Support *133*
8.6.3 NLS Parameters *134*
8.6.4 NLS Views *135*
8.6.5 Linguistic Sorts and Searches *136*
8.6.6 Linguistic Parameters *138*
8.6.7 Linguistic Sort Types *139*

9 MANAGING RESOURCES 140

9.1 DATABASE RESOURCE MANAGER 140
9.1.1 Pending Area *140*
9.1.2 Resource Consumer Groups *141*
9.1.3 User Session Assignments *142*
9.1.4 Resource Plans *143*
9.1.5 Resource Plan Directives *145*
9.2 AUTOMATING TASKS 147
9.2.1 Scheduler *147*
9.2.2 Scheduler Architecture *148*
9.2.3 Administration Tools *149*
9.2.4 Setting Attributes *150*
9.2.5 Scheduler Jobs *151*
9.2.6 Job Classes *151*
9.2.7 Calendaring *152*
9.2.8 Scheduler Windows *153*
9.2.9 Scheduler Views *154*

10 PRACTICE EXAM 155

10.1 QUESTIONS 155

11 ANSWER GUIDE 168

12 REFERENCES 174

13 GLOSSARY 175

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Notice of Rights
All rights reserved. No part of this book may be reproduced or transmitted in any form by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

Notice of Liability
The information in this book is distributed on an “As Is” basis without warranty. While every precaution has been taken in the preparation of the book, neither the author nor the publisher shall have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the instructions contained in this book or by the products described in it.

Trademarks
Many of the designations used by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations appear as requested by the owner of the trademark. All other product names and services identified throughout this book are used in editorial fashion only and for the benefit of such companies with no intention of infringement of the trademark. No such use, or the use of any trade name, is intended to convey endorsement or other affiliation with this book.

Copyright The Art of Service │Brisbane, Australia│Email:service@theartofservice.com
Web: http://theartofservice.com │eLearning: http://theartofservice.org │Phone: +61 (0)7 3252 2055

Write a review to receive any *free* eBook from our Catalog - $99 Value!

If you recently bought this book we would love to hear from you! Benefit from receiving a free eBook from our catalog at http://www.emereo.org/ if you write a review on Amazon (or the online store where you purchased this book) about your last purchase!

How does it work?

To post a review on Amazon, just log in to your account and click on the Create your own review button (under Customer Reviews) of the relevant product page. You can find examples of product reviews in Amazon. If you purchased from another online store, simply follow their procedures.

What happens when I submit my review?

Once you have submitted your review, send us an email at review@emereo.org with the link to your review, and the eBook you would like as our thank you from http://www.emereo.org/. Pick any book you like from the catalog, up to $99 RRP. You will receive an email with your eBook as download link. It is that simple!

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Oracle Database 10g Administration II Certification

The certification for Oracle Database 10g Administration II is a step of the Oracle Certified Professional program. Candidates who have the certification have knowledge in the configuration of an Oracle database for multilingual applications. Additional topics covered by the certification include Oracle technologies and features such as:

- Resource Manager
- Scheduler
- Automatic Storage Management
- VLDB
- Flashback
- Globalization

Oracle Certified Associate for Database Administration is a prerequisite for Oracle training and exams for the Oracle Certified Professional step.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Exam Specifics

Oracle Exams are proctored by Pearson VUE. Scheduling and location of test sites can be obtained at www.pearsonvue.com. Tests are conducted at a testing center. Two valid forms of ID are required when arriving at the center.

Exams are delivered in a secure environment, proctored, and timed.

Specifics about the exam are:

- Exam Number : 1Z0-043
- Price: $125.00 USD
- Time Limit: 90 minutes
- # of Questions: 92
- Question Type: Multiple Choice
- Passing Score: 70%

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Oracle 10g is a comprehensive product for meeting the performance, availability, recovery, and security requirements of all mission-critical applications for an organization.

Oracle 10g

The Oracle Database 10g was released in the spring of 2004. Its primary improvements over previous Oracle databases involved:

- Ease of management
- Enhanced scalability
- Performance management

Ease of Management is obtained through:

- Automatic management of disk storage allocation
- Proactive monitoring of memory structures
- Pre-configured database alerts
- Web-based monitoring and management

Scalability and performance are based on a grid computing model which allows businesses to move away from traditional perspectives of managing several individual servers, each concentrating on a small number of applications and resulting in wasted hardware resources. In this model, the databases a spread across several servers as required with the intent to use hardware resources with the greatest efficiency.

Five editions of Oracle 10g were released:

- Enterprise – All available database features available as bundled or extra-cost options.
- Standard – Includes full clustering and all ease-of-management features.
- Standard Edition One – includes all ease-of-management features.
- Personal – All available database features available as bundled or extra-cost options for individual user databases.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Lite – All database features used in mobile database applications.

Database Concepts

Databases are used to store, manipulate, retrieve, and secure data. While many database solutions are based on flat file, hierarchical file structures, or set connections, Oracle 10g databases utilize a relational model for storing data.

Structured Query Language

The relational model is based on work by Dr. Edgar Codd and published in his 1970 book, "A Relational Model of Data for Large Shared Data Banks." The first implementation of Dr. Codd's work was the creation of computer language, Structured Query Language (SQL), by the IBM Corporation. SQL allows users to interact easily with relational databases without the requirement of complex written computer programs or knowing the physical location of the data on the network.

Common SQL commands include:

- SELECT – display data stored in tables.
- INSERT – add rows to tables.
- DELETE – remove rows from tables.
- UPDATE – modify rows from tables.
- CREATE – add tables
- ALTER – modify tables
- DROP – remove tables
- GRANT - provide access to users
- REVOKE – remove access to users
- COMMIT – makes transaction changes permanent
- ROLLBACK – undo transaction changes

All commands fall into one of four categories for SQL statement type:

- Queries
- Data Definition Language (DDL)

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Data Control Language (DCL)
- Data Manipulation Language (DML)

Transactions are comprised of DML commands. They begin with the first issued DML command and end when changes are made permanent using the COMMIT command or undo any changes with the ROLLBACK command.

Database Constructs

Tables are comprised of columns and rows. The intersection of a single column and single row is called a field. Business data is stored in a collection of tables. These tables are the core construct of a relational database and the driving principle behind Oracle 10g.

When storing business data, a table is created to describe one element of the business. This element will have several attributes that are represented by columns. Each row represents a single entity within the business element. For example, the organizational structure of the business could be constructed of departments. A table can be created to represent the organization. Columns can be added to provide factual information about the different departments. Each row would represent an individual department.

In additional to tables which store business data, Oracle 10g also contains system tables which store data on the database, or metadata. To examine the contents of these metadata tables, administrators can use views. A view is a table, or logical structure, that does not contain any data of its own, but is used to view data from other tables. There are two types of metadata views:

- Data dictionary views
- Dynamic performance views

Metadata Views

Data dictionary views will begin with the prefixes of:

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- DBA_
- ALL_
- USER_

The DBA_ tables will show information on all the tables within the database. The ALL_ tables show all the tables that a particular user owns or has access to. This may not be all the tables in the database. The USER_ table will show only those tables owned by a user.

Depending on the features installed and configured, an Oracle 10g database can contain more than 1,300 data dictionary views, while dynamic performance views will number approximately 350. Most dynamic performance views will be prefixed with V$.

There are several distinctions between data dictionary and dynamic performance views. With dictionary views, DBA_ views usually have plural names and are available only when the database is open and running. The data contained in these views are generally uppercase, static and remains when the database shuts down. Dynamic performance views are usually singular and available when the database is not fully open and running. The data is generally lowercase and is lost whenever the database is shut down.

Relationships and Constraints

A relational database goes beyond simply storing data in tables to building relationships and constraints between tables. A typical Oracle database will have hundreds to thousands of tables. To define how tables are related and even how data should be stored within each table, a set of rules can be created. These rules are called constraints and allow the database designer the ability to enforce business rules on the stored data.

Five types of constraints exist:

- Not Null – values within column must be supplied and do not need to be unique.
- Unique Key – every value in a column must be

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

unique, and null values are allowed.

- Primary Key - every value in a column must be unique, and no null values are allowed.
- Foreign Key – every value in a column must match a value in another column in this or a different table. If no match exists, the value is null. Relationships between two columns in the same table are built using self-referencing foreign keys.
- Check – the value must match a specified value for this column.

Relationships between values in two different tables are established using a foreign key. The design of the relationship is commonly called referential integrity (RI) and is generally enforced using primary key and foreign key constraints. Referential integrity is important as it ensures the relationships rules are enforced as rows are added, deleted, or updated to either table.

Constraints can prevent data to be changed or deleted which would “orphan” other data in the relationship. Tables which have a defined relationship with another table cannot be deleted because of the constraints established. If two tables share a common column, a query can be established which returns rows from both tables.

Segment Types

Tables are just one type of segment within a database. Segments are any entity that consumes physical storage space. The most common segment types are:

- Tables – provides structure for storing data.
- Indexes – improves access to data.
- Rollback – maintains read consistency during user transactions and enables transaction recovery when required.
- Partitions – divides a table into smaller pieces for performance purposes.

Each segment is comprised of contiguous chunks of storage

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

space in the database. This storage space is referred to as an extent. Every segment has at least one extent, and some segments have a number up to 2 billion extents. Each extent is made up of smaller chunks of space called database blocks. The minimum size of an extent is five database clocks. The default size of the blocks can be set at database creation, though multiple block sizes can be found. The common database block sizes are 2KB, 4KB, 8KB, and 16KB.

Each database block is comprised of one or more operating system blocks. The size of the operating system block is generally 512 bytes to 2 KB and dependent on the operating system in use.

Communicating Between Segments

Querying is the most common method of interacting with a database. This method generally utilizes SQL.

Other methods of connecting to a database:

- SQL*Plus
- iSQL*Plus
- Enterprise Management (EM) Database Control
- Oracle Procedural Language for SQL (PL/SQL)

PL/SQL overcomes several limitations present with SQL such as weak condition testing ability, lack of looping capabilities, and lack of exception handling capabilities.

The types of PL/SQL objects used to construct queries include:

- Anonymous Block – not stored in the database, by embedded in a form, web page, or SQL script.
- Procedure – stored in the database and performs a specific action.
- Function – stored in the database and returns a value when called in a SQL statement.
- Package 0- collection of related procedures and functions that performs a related function.
- Trigger – a block of code run whenever an INSERT, UPDATE, or DELETE activity or other defined

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

database events occur.

Oracle Architecture

Database Administrator

Database administrators are tasked with using the available tools to monitor and manage the database. Some of the specific tasks include:

- Provision adequate storage capacity to users
- Enable recovery mechanisms
- Select server hardware where database software will reside
- Installing and configuring Oracle 10g
- Creating databases
- Creating and managing tables and other objects
- Creating and managing database users
- Establishing backup and recovery processes
- Monitoring and tuning database performance

Database Server Architecture

Elements of the Oracle Server Architecture are comprised of:

- User-related processes
- Oracle instances or logical memory structures
- Databases or physical file structures

Two processes allow a user to interact with an instance and databases: user processes and server processes. A user process is initiated when ever a user runs an application. It can exist on the user's computer or application server depending on the technical architecture of the application. When the user process is initiated it will create a connection to the instance. Once the connection is complete a session between the user and the instance is established. At this point, a Server Process on the host server can be started. Server Processes perform the tasks to allow the user to interact with the database.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

For each user, the Program Global Area (PGA) is created which serves as an additional memory structure. The PGA will store user-specific session information.

An Oracle Instance is composed of the System Global Area (SGA), Oracle's main memory structure, and several background processes. When the user assesses the data in the database, the Server Processes are communicating with the SGA.

System Global Area

Three required components and three optional components make up the SGA. They are:

- Shared Pool – caches the most recently used SQL statements issued by users.
- Database Buffer Cache – caches the data more recently accessed by users.
- Redo Log Buffer – stores transaction information for recovery purposes.
- Java Pool – caches the most recently used Java objects and application code.
- Large Pool – caches data for large operations.
- Streams Pool – caches data associated with queued message requests.

The contents of the Shared Pool and Database Buffer Cache are managed by the least recently used (LRU) algorithm. When a Server Process needs to store a statement or block of data into one of these areas, this algorithm is used to identify the least recently used statement or buffer in order to remove and allow the new data to be stored. In this way, those SQL statements and database buffers that are used frequently are maintained in these spaces.

The size of each SGA component can be managed either manually or automatically. When managing manually, the size of the SGA component must be specified, then increased or decreased based on the requirements of the application. When managed automatically, the instance will monitor the utilization of each SGA component and adjust their sizes accordingly. The

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

space of the SGA memory is in chunks called granules, which are dynamically allocated or deallocated based on the demands of the basic components.

Manual or automatic mode for an instance is determined by the settings found in a configuration file – the parameter initialization file. There are two types of parameter initialization files:

- Parameter Files (PFILES)
- Server Parameter Files (SPFILES)

Either type of file can set instance and database configurations, but they work differently. The PFILE name takes on the format of init*instance name*.ora. It can be edited using a text editor and changes to the file will not take effect until the instance is shut down and restarted. PFILE can be created by a SPFILE using the 'create pfile ' from the spfile command. The SPFILE name takes on the format of spfile*instance name*.ora. It cannot be edited directly and changes to the file take effect dynamically. SPFILE can be created by PFILE using the 'create spfile ' from the pfile command.

Basic Initialization Parameters

More than 250 configuration parameters can be documented in the PFILE or SPFILE. These parameters are divided into basic and advanced categories. Oracle recommends that only 30 basic initialization parameters be changed manually.

Here is a listing of initialization parameters:

- CLUSTER_DATABASE - tells the instance that it is part of a clustered environment.
- COMPATIBLE – allows the release level and feature set to be active in the instance to be specified.
- CONTROL_FILES - defines the physical location of the database control files.
- DB_DLOCK_SIZE – specifies the default database block size.
- DB_CREATE_FILE_DEST – identifies the directory location where database datafiles will b e created using the Oracle Managed Files feature

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- DB_CREATE_ONLINE_LOG_DEST_n – identifies the locations where database log files are created using the Oracle Managed Files feature.
- DB_DOMAIN – identifies the logical location of the database on the network.
- DB_NAME – identifies the name of the database mounted by the instance.
- DB_RECOVERY_FILE_DEST – identifies the location fro recovery files using the Flash Recovery feature.
- DB_RECOVERY_FILE_DEST_SIZE – identifies the amount of disk space available to store Flash Recovery files.
- DB_UNIQUE_NAME – identifies a globally unique name for the database within the enterprise.
- INSTANCE_NUMBER – identifies the instance in a Real Application Clusters (RAC) environment.
- JOB_QUEUE_PROCESSES – identifies the number of background processes started to handle jobs submitted through the Enterprise Manager or DBWS_JOBS.
- LOG_ARCHIVE_DEST_n – identifies where archived redo log files are written.
- LOG_ARCHIVE_DEST_STATE_n – defines how specified locations are used for log archiving.
- NLS_LANGUAGE – identifies the default language of the database.
- NLS_TERRITORY -identifies the default region or territory of the database.
- OPEN_CURSORS – sets the maximum number of cursors allowed to be open simultaneously.
- PGA_AGGREGATE_TARGET – allots the overall amount of memory that all PGA processes can consume.
- PROCESSES – identifies the maximum number of operating systems processes that can connect to the instance.
- REMOTE_LISTENER – identifies a network name that points to the address or list of addresses of remote Oracle Net users.
- REMOTE_LOGIN_PASSWORDFILE – defines if an

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

instance uses a password file and what type.

- ROLLBACK_SEGMENTS – identifies if the Automatic Undo Management is not being used.
- SESSIONS – defines the maximum number of sessions that can connect to the database.
- SGA_TARGET – defines the maximum number size of the SGA.
- SHARED_SERVERS – defines the number of Shared Server processes to start when an instance is started.
- STAR_TRANSFORMATION_ENABLED – defines whether the optimizer will consider star transformations when executing queries.
- UNDO_MANAGEMENT – defines whether the system undo is automatic or manually managed.
- UNDO_TABLESPACE – identifies which tablespace stores undo segments if using the Automatic Undo Management option.

Oracle Background Processes

Oracle Background Processes perform specific jobs for managing the interface. Five of the processes are required and found in all Oracle instances. They are:

- System Monitor (SMON) – performs instance recovery following an instance crash, coalesces free space and managing space used for sorting.
- Process Monitor (PMON) – will clean up failed user database connections.
- Database Writer (DBWn) – writes modified database blocks from the Buffer Cache to the datafiles on disk.
- Log Writer (LGWR) – writes transaction recovery information from Redo Log Buffer to online Redo Log files on disk.
- Checkpoint (CKPT) – updates database files following a Checkpoint Event.

Several background processes are optional and their use is dependent on the features used in the database. They include:

- Archiver (ARCn) – copies transaction recovery

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

information written by the LGWR to the online Redo Log files and a secondary location in case of recovery.

- Recoverer (RECO) – recovers failed transactions distributed across multiple databases when using the distributed database feature.
- Job Queue Monitor (CJQn) – assigns to the Job Queue processes when using the job scheduling feature.
- Job Queue (Jnnn) – executes database jobs scheduled using the job scheduling feature.
- Queue Monitor (QMNn) – used to carry out portions of a larger query when using the Parallel Query feature.
- Dispatcher (Dnnn) – assigns database requests to a queue to be serviced by Shared Server processes.
- Shared Server (Snnn) – Server Processes that are shared across several users.
- Memory Manager (MMAN) – Manages the size of individual SGA components when using the Automatic Shared Memory Management feature.
- Memory Monitor (MMON) – gathers and analyzes statistics used by the Automatic Workload Repository feature.
- Memory Monitor Light (MMNL) - gathers and analyzes statistics used by the Automatic Workload Repository feature.
- Recovery Writer (RVWR) – writes recovery information to disk when using the Flashback Database Recovery feature.
- Change tracking Writer (CTWR) – keeps track of the database clocks changed when using the Incremental Recovery Manager feature.

Oracle Database

The database is a set of physical files residing on the host server's disk drives. These files are called:

- Control files – identifies the location of other physical

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

files and defines the database name, block size, character set, and recovery information.
- Data files – all application data and internal metadata.
- Redo files – record of all changes made to the database.

Control files are critical components of the database and they hold important information not available elsewhere in the database. The information they store includes the database name, the names, locations, and sizes of the datafiles and redo log files, and information used to recover the database in the case of disk failure or user error. They are created when the database is created in locations specified in the control_files parameter in the parameter file. The loss of a control file can have a negative effect on the recovery of the database, so many administrators multiplex them to multiple locations. The CKPT process is used to automatically update each file as required and ensures multiple copies of the files are synchronized.

Datafiles store the actual data inserted into each table in the database. The size of the datafile is proportional to the amount of data they store. They also make up the physical structure behind another a logical database storage area called a tablespace. Tablespaces allow related segments to be logically grouped together. Every 10g database must have at least three tablespaces:
- SYSTEM – stores the data dictionary tables and PL/SQL code.
- SYSAUX – stores segments used for database options.
- TEMP – used for performing large sort operations and required when the SYSTEM tablespace is created as a locally managed tablespace.

Some common tablespaces include:
- TOOLS – used to store segments for nonapplication management tools.
- USERS – used as the default tablespace for database users.
- UNDOTBS1 – used to store transaction information for read consistency and recovery purposes.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Each tablespace must have at least one associated database file.

For each transaction performed in a database, the information needed to reproduce the transaction in the event of a failure is automatically recorded in the redo Log Buffer. The contents of the buffer are then written to the redo log files using the LGWR process. To ensure availability for recovery, the redo log files are typically multiplexed or copied. Sets of redo logs are referred to as redo log groups with each multiplexed files within the group referred to as a redo log group member.

Other files that are associated with the Oracle database but not technically a part of it are:

- Password files – stores names of users granted SYSDBA and SYSOPER privileges.
- Parameter files – PFILE and SPFILE files
- Archived redo log files – copy of the contents of previous online redo logs.
- Oracle Net – entries for configuring database listener and client-to-database connectivity.

Installing Oracle 10g

Review Documentation

Before installing Oracle 10g, several documents should be reviewed to understand the installation requirements. The primary documents include:

- Installation guide for the operating system
- General release notes for Oracle version
- Operating-system-specific release notes for the Oracle version
- Quick start installation guides

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

System Requirements

Server hardware specifications should be reviewed to verify if they meet or exceed the specifications listed in the installation documents. Four areas are minimally required that specifications meet criteria set forth:

- Release level of operating system
- Memory capacity to perform the install and run an instance
- CPU resources to perform the install and run an instance
- Storage space to perform the install and run an instance

The minimum hardware requirements for Oracle 10g, include:

- 512MB memory
- 1GB swap space
- 400MB free space in /tmp directory for Unix systems
- 1.5GB free disk space for base Oracle 10g installation and 1GB of disk space is needed to create a database using the Database Configuration Assistant.

Unix kernel parameters should be examined for Unix installation since configurations of Unix operating system settings for system-level operations can impact Oracle related activities.

Installation Plan

Installation planning is simplified by adopting the Optimal Flexible Architecture (OFA) model for managing installations. The OFA model addresses the following four areas:

- Naming conventions for file systems and mount points
- Naming conventions for directory paths
- Naming conventions for database files
- Standard locations for Oracle-related files.

On Unix systems, every file is owned by an operating system user account and a Unix user account must be created for the

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

user who will own the Oracle binaries. The user name can be anything. An operating system group should be created for the Oracle user.

Files on a Unix system are stored on logical storage areas, or volumes, that are mounted to directories called mount points. These volumes and mount points should be named based on recommendations from the OFA.

OFA Directory Paths should use a consistent and meaningful naming convention. The model also assigns standard operating system environment variable names to some of the directory paths as 'nicknames.'

Oracle Universal Installer

To install and configure the Oracle 10g software, use the Oracle Universal Installer (OUI). The OUI provides the same installation aesthetics no matter what operating system the database will be installed on, because its programming is based in the Java language. The OUI process involves six primary steps:

- Mounting the CD and starting the OUI.
- Performing preinstallation checks.
- Responding to server specific prompts for database settings.
- Copying files from install media to $ORACLE_HOME.
- Compiling the Oracle binaries.
- Performing post-install operations using Configuration Assistants.

The installation process starts when the Oracle 10g CD in the server. In Unix systems, an operating system command may be required to mount the CD to the server. The following environment variables should be set:

- $ORACLE_BASE
- $ORACLE_HOME
- $PATH
- $LD_LIBRARY_PATH
- $DISPLAY (UNIX)

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

The command, runInstaller.sh, can be used to start the OUI. At this point, the preinstallation checks will be performed. These checks will run tests on the system version, available RAM, kernel settings, and other system components to ensure successful installation and running of the product.

After the checks, several prompts will appear asking for configuration information about the installed products. The requests include:

- The location for inventory files.
- The name of the operating system group.
- The type of installation to perform.
- Creating a database after install.

During these prompts, additional checks may be performed. After providing the requested information, the copy process starts indicated by a progress bar. Completing the file copy portion of the installation, the binaries are linked to create the executable files needed to run Oracle 10g. A subsequent progress bar is shown. Unix administrators will be prompted to execute a second configuration script as the superuser root.

Some post-installation configuration activities will be performed to complete the installation process.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Oracle Database

Management Framework

The Oracle Enterprise Management Framework is a comprehensive set of integrated tools used to perform database tasks efficiently and monitor the database components across the enterprise effectively. The framework is customizable to meet the business requirements through a web-based interface. The framework can also be extended using the software developer's kit (SDK).

The Oracle Enterprise Management Framework has several functional areas:

- Managed Targets – using the Enterprise Manager, can add and removed targets, such as databases, application servers, web servers, applications, and agents as the enterprise changes.
- Management Services – a Java-based web component serving as the interface used to monitor and control managed targets.
- Oracle management Repository – holds configuration and monitoring information about managed targets. Includes two tablespaces containing information about administrators, targets, and applications.
- Oracle Enterprise Manager 10g Grid Control – a web-based user interface used to centrally managed and communicate with all database components within the enterprise, specifically hosts, databases, listeners, application servers, web servers, and web applications.
- Oracle Enterprise Manager 10g Database Controls – a web-based interface for monitoring and administrating a single database instance or RAC environment.
- Application Server Control – a web-based interface

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

used to monitor and administer a single Application Server instance, multiple Application Server instances, or Application Server Clusters.

Oracle Management Agents

Two methods for administrating databases within the enterprise include centrally managing all the databases using the Grid Control or managing each database individually using the Database Control. In either case, management agents may need to be started and stopped. When using the Grid Control, an agent process must be started for each managed target server available to the framework.

Oracle Management Agents provide the mechanisms for communication between all targets of the framework. They work in a background process that runs on a managed target server. These agents identify and collect data about the entities within the framework and store that data within the Management Repository.

To start or stop a management agent, use the EMCTL command-line utility. Specific commands include:

To start agent	emctl start agent
To stop agent	emctl stop agent

Managing individual databases individually uses the Enterprise Manager Database Control web tool. To open this tool, issue the following command:

```
emctl start dbconsole
```

To stop the tool, issue the command:

```
emctl stop dbconsole
```

The status of a specific agent can be viewed by issuing the command:

```
emctl status dbconsole
```


Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Oracle Database Configuration Assistant (ODCA)

The Oracle Database Configuration Assistant (DBCA) is a Java-based tool used to create and manage Oracle databases. The DBCA can be accessed as an application in a Windows environment or from the command line in Unix. In Windows, the tool can be opened from Start > Programs > Oracle Home > Configuration and Migration Tools > Database Configuration Assistant. Command line access from either Unix or Window uses the dbca command from the $ORACLE_HOME/bin location.

The options available after opening the DBCA:

- Create a Database
- Configure Database Options
- Delete a Database
- Manage Templates

When creating a database, a global database name and Oracle system identification name (Oracle SID) can be set. The global database name is a fully qualified name of the database in the enterprise that must be unique within the network domain. It is comprised of the database name and domain in the format: database_name.database_domain. The SID is the name of the instance associated with the database which must be unique on the server. It must be a maximum length of eight characters.

Database Templates

Several preconfigured database templates are already available with the DBCA. These documents are XML-based and contain information necessary to create an Oracle database. A custom database definition can be created as a template for later use as well. The predefined databases templates are designed to optimize work for particular type of workload and include:

- Data Warehouse
- General Purpose
- Transaction Processing

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

The database definitions for these templates can be viewed by clicking Show Details in the DBCA. Each section provides information about the template. Specific sections include:

- Common Options – identifies which options can be installed.
- Initialization Parameters – displays the common parameters and settings.
- Character Sets – displays the character sets to be used.
- Control Files – displays the control filenames and locations.
- Tablespaces – displays tablespace names and types.
- Datafiles – displays the datafile names for each tablespace and their size.
- Redo Log Groups – displays the group number and size.

Database Credentials

Passwords for different user accounts set up automatically when configuring databases by using the Database Credentials screen. The same password can be used for all critical accounts or have a different password for each of the accounts. The four accounts preconfigured when a database is setup include:

- SYS – has ownership of all the internal Oracle tables that complete the data dictionary.
- SYSTEM – provides support for additional administrative tables and views.
- DBSNMP – a login used by the Enterprise Manager facility to monitor and gather performance statistics about the database.
- SYSMAN – the equivalent of the SYS user for the Enterprise Manager facility to create and modify other Enterprise manager administrator accounts.

Storage Options

To configure the disk storage areas used by the database, there

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

are three choices available in the Storage Options screen:

- File System
- Automatic Storage Management (ASM)
- Raw Devices

The most common type of storage configuration for database existing before Oracle 10g is the file system storage. This storage definition relies on the underlying operating system to maintain and manage the actual files defined by the DBA. A set of datafile names and directory locations will be suggested by the DBCA that can be used and modified later. The driving force behind these suggestions is the Optimal Flexible Architecture.

The new type of storage mechanism available to Oracle 10g exists called Automated Storage Management (ASM). It is designed to take the burden of disk and storage management and place it on the Oracle install. Instead of managing individual databases, they are placed into defined disk groups for file management. One or more disk group can be defined as a logical unit that can be viewed as a single unit. The storage definitions of a database can be managed within a second database to track disk group allocation using ASM. By selecting the ASM option, a series of screen guides will appear to complete the process of defining the secondary ASM database instance.

Raw devices are disks that are not managed by the underlying operating system, but made directly to the underlying hardware. The systems administrator will predefine the raw disk partitions for the specific raw devices, while the DBA will map those devices to specific datafiles and redo log files.

After choosing the storage type to use, the location where the files that make up the database must be defined. The type of storage option chosen will increase or decrease the number of location options available. The three options provided in the Database File Locations screen are:

- Use Database File Locations From Template
- Use Common Location For All Database Files
- Use Oracle-Managed Files

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Any predefined database template will define the location that should be used by Oracle when creating a new database, though the filenames and locations can be reviewed and modified if needed. When using the option for a common location, a new directory can be specified for all database files, though filenames and locations can be modified later in the database definition process when creating new databases. ASM will provide the Use Oracle-Managed File option and require the directory path to the disk area for storage to be selected.

Recovery Configuration

Within the Recovery Configuration screen, the backup and recovery strategy for the database can be set up. A new option for recovery is the Oracle Flash Recovery and is the foundation for the new Automated Disk-Based recovery feature. This option is designed to simplify backups by providing a centralized location for maintaining and managing all files related to database backups.

An area of the disk is dedicated to the storage and management of files needed for database recovery called the Flash Recovery Area. This area is separate from the other components of the Oracle database that contain datafiles, redo log files, and control files. The area is used to store and manage archive logs and used by the Enterprise Manager and the Oracle Recovery Manager (RMAN).

The location and size of the disk area for the Flash Recovery area can be specified. The default location of the directory is ORACLE_BASE\flash_recovery_area. To view a summary of Oracle file location parameters, click the File Location Variables option in the Recovery Configuration screen. The default size of the Flash Recovery area is 2048MB.

Archive logging is a feature that can be enabled in the Recovery Configuration screen. This option is a mechanism that allows point-to-failure recovery of a database to be performed. Additional parameters can be enabled for archive mode.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Database Content

When creating a custom database in the Database Template screen, a Database Content screen will be presented. Here, the database components to be installed can be chosen, including:

- Oracle Text – support for multimedia content.
- Oracle OLAP – facilities for creating and deploying online analytical processing applications.
- Oracle Spatial – provides the components and infrastructure for Oracle to manage and maintain geographic and spatial information.
- A set of analytical tools and extended algorithms to facilitate data mining.
- Oracle Ultra Search – capabilities to perform extended text and searches within the Oracle database.
- Oracle Label Security – manages and controls access to sensitive information within the database.
- Sample Schemas – working examples of how to configure and use extended features of the database.
- Enterprise Manager Repository – specifies the location of the schema used to manage the content.

Examples of working databases can be installed using the DBCA, including:

- Human Resources
- Order Entry
- Product Media
- Sales History
- Queued Shipping

Initialization Parameters

The size and setup characteristics of the Oracle instance can be configured using the Initialization Parameters screen. Four tabs are included in this screen:

- Memory
- Sizing
- Character Sets

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Connection Mode

The Memory tab controls the size of the database parameters defining the overall memory footprint of an Oracle instance. Two general approaches are available for managing these parameters: allow Oracle to set and manage most parameters, or customize each of the initialization parameters for the database.

When allowing Oracle to manage the memory, the system will allocate the memory to various components within the SGA and PGA. The memory allocation is automatic and becomes a percentage of the overall physical memory available on the server. The default is 40 percent of the total memory available. The Show Memory Distribution button allows the DBA to see how Oracle will allocate the memory between the SGA and PGA.

Customizing memory requires the DBA to configure memory in the following areas:

- Shared Pool
- Buffer Cache
- Java Pool
- Large Pool
- PGA Size

The Sizing tab can be used to configure the block size of the database and the number of processes that connect to the database. The Block Size setting refers to the smallest unit of storage within the database. The storage of all database objects depends on the block size which maintains a modifiable default of 8KB. This setting cannot be modified after creating a database. The operating system can determine the minimum and maximum size of a block. While 8KB is sufficient for transaction-oriented applications, larger block sizes allow data warehouse-type applications.

The maximum number of simultaneous operating system processes can be set in the Sizing tab as well. At least six processes for each of the background processes must be accounted for.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Within the Character Set tab, the DBA can set the character set, the national character set, default language, and the default date format. The character set used will define the encoding scheme used to determine how characters are displayed and stored within the environment. A character set determines the languages represented in the environment and even how the database interacts with the operating system and how much storage is required. The default character set is based on the language setting of the operating system.

The Connection Mode tab will specify which types of connections are available to the database. Two modes can be chosen from: dedicated server (default) or shared server.

Database Storage

After the Initialization Parameters screen, the DBA will be provided an opportunity to review and change the location of objects that create the database. The details of each component can be displayed. New objects can be added to a particular group when not using a template to create a database.

Creation Options

Once the DBA has moved through the options provided, two options are left to determine what is created:

- Create Database – the DBCA immediately create the database.
- Save as a Database Template – the database definition is saved to a template and the database is created immediately or the template is used to create the database later.

If creating the database immediately is the option chosen, a Configuration screen will be displayed that summarizes the database configuration options chosen, including

- Installed options
- Initialization parameter settings
- Character set settings

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Datafile names and locations
- Redo log filenames and locations
- Control filenames and locations

Other DBCA Activities

Other activities can be performed using the DBCA:

- Configuring a Database
- Deleting a Database
- Managing Templates

Configuring the Database

Various aspects of an existing database can be changed using the DBCA. To change configuration, select the Configure Database Options in the DBCA Operations screen. If the database has not already started, the DBCA will automatically start it. To connect to the database to change configurations, DBA authority is required.

Some options that were not previously included in the database can be added. The server type can also be changed.

Deleting the Database

A database can also be deleted from the DBCA Operations screen. The DBCA will remove all the files associated with the database from the disk. In Windows, the DBCA will remove all the services associated with the database.

Managing Templates

Two types of templates are used by the DBCA:

- Seed
- Nonseed

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Seed templates are definitions that contain database definition information and actual datafiles and redo log files. These prebuilt datafiles include all the schema information needed to make the database creation process faster. Seed templates have .dbc extensions and the associated redo logs and datafiles are stored as files with the .djf extension. Using the seed template, the database name, locations, number of control files and redo log groups, and initialization parameters can be changed.

Nonseed templates contain custom-defined database definitions. There are no preconfigured datafiles and redo logs. Nonseed templates carry a .dbt extension.

Template definitions can be created by:

- Creating from an existing template.
- Creating structure from an existing database.
- Creating structure and data from an existing database.

Existing templates can also be deleted using the DBCA.

Parameter Initialization Files

Parameter Initialization files are used to store information about initialization parameters used when an Oracle instance starts. A parameter file is either a plaintext file (PFILE) or binary parameter file (SPFILE).

When a startup command is issued, Oracle will use a search hierarchy to specify either a PFILE or SPFILE. The first parameter file searched for is the spfile$ORACLE_SID.ora. If not found, the spfile.ora file is searched for. The final file to be searched is a PFILE with the default name, init$ORACLE_SID.ora.

If the files are not in the default location or other parameter files should be used to start the database, the appropriate parameter file should be included in the startup command.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

To change the initialization parameters, the EM Database Control tool can be used or they can be changed manually. To use the tool, navigate to the Administration menu of the EM Database Control tool to modify existing database parameters.

The Initialization Parameters screen has two tabs: Current and SPFile. The Current tab displays the active settings for initialization parameters for the database instance. If the parameter is marked Dynamic, it can be modified and the changes are effective immediately. How they are also temporary and the original values will be used the next time the database is stopped and started.

If a SPFILE is being used, a SPFILE tab will be displayed. This tab allows existing database parameters to be changed. The difference in changes in this tab and the Current tab is that the changes will be permanent across database startups and shutdowns. Changes can be applied to the SPFILE only or to the file and current instance.

Start and Stop Databases

Administrative Authorization

To start up and shut down an Oracle instance, the proper administrative privileges are required. Two connection account authorizations are present to perform these tasks: SYSDBA and SYSOPER.

SYSDBA authorization allows any database task to be performed and is only able to connect to the database in the SYS schema when the database is initially installed. This authorization and the SYSOPER authorization can be granted to others without connecting as the SYS user.

SYSOPER authorization has fewer privileges than SYSSYS such as access to non-administrative tasks in the database. IT does however allow startup and shutdown activities to be

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

performed.

Startup Modes

When a database starts up, it will pass through three modes:

- NOMOUNT
- MOUNT
- OPEN

STARTUP NOMOUNT will start the instance without mounting the database. In this mode, the parameter file is read, background processes and memory structures are initialized, but none are attached or communicating with the disk structure of the database. The database is not available for use. The NOMOUNT mode is useful for several administrative tasks, the most common being the running of a script to create the underlying database.

If Oracle has problems accessing the control file structure because they are damaged or unavailable, a database may not be able to move from the UNMOUNT mode to the MOUNT mode. If this occurs, or something similar, the problem needs to be resolved before the process can continue.

STARTUP MOUNT is similar in activities to the previous mode except that all components are attached and communicate with the disk structure of the database. To make this connection, Oracle uses information provided by the control files. This mode allows recovery to be performed, as well as other administrative tasks such as physical change of file locations or placing the database in archive mode.

STARTUP OPEN is the default startup mode if no other mode is specified in the startup command line. All the steps performed in the NOMOUNT and MOUNT modes are performed here. The different in this mode is that the database is now available to users.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Special Startup Modes

Some special startup modes may be used instead of the NOMOUNT, MOUNT, or OPEN modes. They are:

- STARTUP FORCE
- STARTUP RESTRICT

STARTUP FORCE is typically used when starting the database in normal fashion is difficult. One common trigger for such an event is when power is unexpectedly lost and the database stopped without going through the shutdown process. This puts the database in a state where STARTUP FORCE has to be used if the normal startup modes do not work. If can be used no matter what mode the database is or was currently it as it works by performing a shutdown abort and restarting the database.

STARTUP RESTRICT will startup the database in OPEN mode, but will only grant database access to users who have the RESTRICTED SESSION privilege. This special mode is generally used when performing maintenance on the database while it is open while ensuring that general users cannot connect and perform work on the database. The mode also allows database exports and import to be performed without worrying about user activity interfering. The restricted session can be disabled afterwards to allow all users to have access to the database.

EM Database Control Startup

To startup the database using the EM Database Control tool, click the Startup button. An operating system username and password is required as well as an Oracle user ID with SYSDBA or SYSOPER privileges and password. Click OK after supplying all the credentials.

At this point, the DBA can continue the startup, abort, or choose Advanced Options for the startup. The Advanced Options allows:

- A different mode to be chosen.
- Parameter files to be chosen.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

SQL*Plus Startup

The SQL*Plus command line facility can be used to startup the database by any user with the SYSDBA or SYSOPER privileges. The syntax of the startup command is:

STARTUP [NOMOUNT|MOUNT|OPEN] [PFILE/SPFILE=] [RESTRICT]

Database Shutdown Processes

There are several processes for shutting down a database, including:

- NORMAL
- TRANSACTIONAL
- IMMEDIATE
- ABORT

The default process for shutting down a database is NORMAL. With this process:

- New Oracle connections are not allowed from the time the shutdown command is issued.
- The process will proceed when all users are disconnected from the database. IT will wait until this is done.

Because all the users have to be disconnected before shutting down in this process, the wait may be indefinite, especially if a user is still connected but has left for the day. Additional work is required to identify the user and force a disconnection. The NORMAL shutdown is also referred to as a 'clean' shutdown since no recovery is required when the database is started again.

A TRANSACTIONAL shutdown is more aggressive than a normal shutdown with the following characteristics:

- New Oracle connections are not allowed after the shutdown command is issued.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- New transactions are not allowed to start after the shutdown command is issued.
- Once all active transactions on the database are completed, all client connections are disconnected.

By waiting until the transactions are complete, the users connected to the database will not lose any work. This process of shutting down is also considered clean and does not require any recovery when the database is started again.

The IMMEDIATE shutdown process is even more aggressive and has the following characteristics:

- New Oracle connections are not allowed after the shutdown command is issued.
- Uncommitted transactions are rolled back, causing possible work loss.
- Any unfinished transactions are rolled back and users are immediately disconnected.

This process is generally used when shutdown must be performed unattended or as part of a script. This shutdown process is also considered clean despite the rolling back of transactions and disconnection of users. It is clean because no recovery activity is required.

The SHUTDOWN ABORT process is the most aggressive:

- New Oracle connections are not allowed after the shutdown command is issued.
- Any SQL statements in progress are terminated.
- Uncommitted work is not rolled back.
- All user connected are immediately terminated.

This process should not be used on a regular basis but only when other shutdown processes have failed. This is not a clean shutdown and recovery is required the next time the database starts.

EM Database Control Shutdown

To shutdown the database in the EM Database Control, click the

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Shutdown button in the General section of the tool. Credentials have to be provided. The default shutdown process is SHUTDOWN IMMEDIATE, but different options are provided through the Advanced Options button.

SQL*Plus Shutdown

Using the SQL*Plus command line facility, shutting down a database uses the following syntax:

SHUTDOWN
[NORMAL|IMMEDIATE|RESTRICT|ABORT]

Alert Log Monitoring

The Database alert log contains information about certain events and errors that occur within the database. The log is chronological. The information can be used to diagnose system problems and review event histories, such as:

- Startup and shutdown information.
- Certain types of administrative actions.
- Certain types of database errors.
- Values of initialization parameters.

The location of the alert log is governed by the parameter, BACKGROUND_DUMP_DEST.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Storage Structures

Disk space management within Oracle creates a hierarchy with tablespaces being the highest level. Moving through the hierarchy:

- Segments reside in only one tablespace.
- One or more extents make up each segment.
- Each extent in constructed by a contiguous set of data blocks.
- A data block is a fixed number of bytes of disk space.

Each extent can reside on only one datafile. For a segment to straddle multiple datafiles, in must be constructed from multiple extents located in separate datafiles

The data block is the lowest level of disk space management in Oracle. Its size is a tablespace attribute. While SYSTEM and SYSAUX tablespaces maintain a standard data block size defined at database creation, other tablespaces can have different data block sizes which are defined at tablespace creation. A single database can have as many as five data block sizes. The sizes typically range from 8KB to 32KB. It must be a multiple of the physical clock size of the storage devices.

Tablespace Management

Tablespaces bridge physical structures and logical structures. They can store zero or more segments which are schema objects requiring storage outside of the data dictionary. Tables and indexes are examples of segments. Constraints and sequences are examples of schema objects that store data within the data dictionary and are not considered segments.

Tablespaces can be used to manage data from tables and indexes easily. A tablespace can be taken offline and recovered separately from the rest of the database. Tablespaces also allow tables and indexes to be physically separated onto disk

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

drives without using a logical volume manager. This can be done by placing tables in a tablespace that sits on datafiles residing on one set of disk drives while placing indexes for those tables into another tablespace that sits on datafiles on another set of disk drives.

There are several default tablespaces are always created when a new database is create:

- SYSTEM tablespace is used for the data dictionary and not for schema objects.
- SYSAUX tablespace is a new Oracle 10g feature and stores schema objects associated with other Oracle features.

One or more temporary tablespaces are created in a database. THE SYSTEM and SYSAUX should not be used as the default tablespaces for user sand applications.

Creating Tablespaces

Tablespaces are created when either the CREATE DATABASE or CREATE TABLESPACE statements are used. Several choices must be made when creating tablespaces:

- Creating Bigfile or Smallfile tablespaces.
- Managing extents locally or with the dictionary.
- Automatic or manual management of segment space.

Bigfile tablespaces are built on a single datafile which can be as large as 2^{32} data blocks in size (8KB data blocks create a 32TB tablespace). As such, bigfile tablespaces are intended for very large databases. Such a tablespace may have thousands of datafiles, though the fewer the number, the faster the overall operations. To create a bigfile tablespace, insert the BIGFILE keyword in the CREATE statement.

Smallfile tablespaces utilize up to 1,022 datafiles that are 2^{22} data blocks in size (8KB data blocks create 32GB datafiles). If the maximum number of datafiles is used with the maximum clock sizes, the total database will be slightly less than 32TB or

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

the same size as a bigfile tablespace. To create a smallfile tablespace, either omit the BIGFILE keyword or use the SMALLFILE keyword explicitly.

To ease the administration of files used by a database, Oracle Managed Files (OMF) can allow operations to be specified in terms of tablespaces and not operating system files. To enable OMF, set the initialization parameter DB_CREATE_FILE_DEST to the directory where the database should create and manage data and temp files. These files will be named by the database and not explicitly by the DBA. To create a tablespace using OMF, omit the filename.

Extent management can be done locally or through the dictionary. Using the older technique of the dictionary, the database tracks free and used extents in the data dictionary. The FET$ and UET$ tables are changes with the recursive SQL. Managing the extents locally, they are tracked using bitmaps and eliminating the need for recursive SQL.

Two options are available in a locally managed tablespace for allocating extents: UNIFORM or AUTOALLOCATE. The uniform option has the database allocate and de-allocate extents with the same specified size. The default size if unspecified is 1MB. This option is not available for undo tablespaces but is the default for temporary tablespaces. To create local uniform extents of a specified size use the clause EXTENT MANAGEMENT LOCAL UNIFORM SIZE X in the CREATE TABLESPACE statement where X is the size desired.

AUTOALLOCATE allows the database to vary the size of extents for each segment. This allows small segments to remain small while large segments can grow without obtaining too many extents. The best used of the AUTOALLOCATE option is for general-purpose mixture of small and large tables. To create tablespaces using the Autoallocate option, use the clause EXTENT MANAGEMENT LOCAL AUTOALLOCATE in the CREATE TABLESPACE statement.

Local extent management use requires segment space management which must be set to manual or automatic. Manual

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

segment space management exists for backward compatibility. It uses free block lists to identify the data blocks available for inserts. A clock made available for insert is controlled using PCT_FREE and PCT_USED. After each INSERT or UPDATE, the database will compare the remaining free space in that data block with the PCT-FREE setting. If the data block is less than the PCT_FREE free space, the data block is removed from the free block list making it unavailable for inserts. The remaining free space is reserved for update operations that may increase the size of rows in the data block. After each INSERT or DELETE, the database will compare the used space in the data block with the PCT_USED setting. If the data block is less than the PCT_USED used space, the data block is considered empty enough for inserts and placed in the free block list.

When specifying manual segment space management, used the SEGMENT SPACE MANAGEMENT MANUAL clause of the CREATE TABLESPACE statement, or omit the SEGMENT SPACE MANAGEMENT clause. The default for Oracle 10g is MANUAL, although recommendations from Oracle suggest using AUTOMATIC.

With automatic segment space management, bitmaps are used instead of free lists to identify available data blocks for inserts. The parameters PCT_FREE or PCT_USED are ignored. This option is not available for temporary or system tablespaces. To specify automatic segment space management, use the SEGMENT SPACE MANAGEMENT AUOT clause in CREATE TABLESPACE.

Other Tablespaces

A temporary tablespace is sued for temporary segments and created, managed, and dropped by the database as they are required. They are typically created during sorting operations such as ORDER BY, GROUP BY, and CREATE INDEX and other operations such as hash joins and inserts into temporary tables.

A temporary tablespace can be created at database creation

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

with the DEFAULT TEMPORARY TABLESPACE clause of the CREATE DATABASE statement or after the database has been created with the CREATE TEMPORARY TABLESPACE statement. Temp files are used instead of datafiles. They are only available with temporary tablespaces, are never backed up and do not log data changes in the redo logs. Though datafiles are completely allocated and initialized at creation time, temp files may not be allocated the disk space specified.
Undo tablespaces store undo segments allowing databases to:

- Roll back a transaction explicitly with a ROLLBACK statement.
- Roll back a transaction implicitly.
- Reconstructing a read-consistent image of data.
- Recovering from logical corruptions.

To create an undo tablespace at database creation time, set the initialization parameter UNDO_MANAGEMENT=AUTO and include the UNDO TABLESPACE clause in the CREATE DATABASE statement. After database creation, create an undo tablespace using the CREATE UNDO TABLESPACE statement.

Removing and Modifying Tablespace

To remove a tablespace from a database, use the DROP TABLESPACE statement. Adding the clause, INCLUDING CONTENTS, will recursively remove any segments in the tablespace. Removing a tablespace does not automatically remove the datafiles from the file system and requires the additional clause INCLUDING CONTENTS AND DATAFILES.

To modify the attributes of a tablespace use the ALTER TABLESPACE statement. Attributes that can be changed are:

- Tablespace name
- Adding datafile to smallfile tablespace
- Switching a tablespace offline or online
- Making a tablespace read-only or read/write
- Changing the user-managed backup state

To rename a tablespace, use the ALTER TABLESPACE statement with a RENAME clause. This can also be used to

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

move a tablespace.

To add a datafile, use the ALTER TABLESPACE statement with an ADD clause.

To take a tablespace offline, use the ALTER TABLESPACE statement with an OFFLINE clause.

To bring a tablespace back online, use the ALTER TABLESPACE statement with an ONLINE clause.

To make a tablespace read-only, its data must be static data. If read-only, the tablespace will no be backed up in the nightly or weekly database backups. Use the ALTER TABLESPACE statement with an READ ONLY clause.

To make a tablespace read/write, use the ALTER TABLESPACE statement with a READ WRITE clause.

To perform backups outside of the Recovery Manager tool, the tablespace must be place in backup mode. To do this, use the ALTER TABLESPACE statement with a BEGIN BACKUP clause. The tablespaces need to be taken out of backup mode or problems will appear. To exit backup mode, use the ALTER TABLESPACE statement with an END BACKUP clause.

Tablespace Information

The EM Database Control tool can be used to view information on tablespaces, including:

- DBA_TABLESPACES
- DBA_DATA_FILES
- DBA_TEMPT_FILES
- V$TABLESPACE

The DBA_TABLESPACES view displays each tablespace in the database as a single row and includes the information:

- Block size
- Status: online, offline, or read-only
- Contents: undo, temporary, or permanent

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Dictionary or locally managed extents
- Automatic or manual segment space management
- Bigfile or smallfile tablespace

This information is available using a command line statement SELECT query.

Tablespace information can also be obtained by generating a DDL to re-create the tablespace. Use the PL/SQL package DBMS_METADATA to generate this DDL. The output is a CREATE TABLESPACE statement that contains all the attributes for the tablespace.

The interactive GUI tool in the EM Database Control can be used to monitor and manage database structures. To use the GUI tool:

- Set the browser to the EM URL for the database.
- Log into the database.
- Navigate to the administration tab.
- Click the Tablespaces link under Storage.
- Click the radio button of the tablespace to work with and click Edit.
- Navigate to the General, Storage, and Thresholds edit screens.

Datafile Management

When not using Oracle Managed Files (OMF), datafiles must be managed manually. In this situation, the database will create or reuse one or more datafiles in the sizes and locations. A datafile can belong to only one tablespace and one database at a time. Temp files are a special datafile used in temporary tablespace. When a database creates and reuses a datafile, the operating system file is allocated and initialized. The temp file will not be initialized.

The operations that can be performed on a datafile are:

- Changing the size
- Taking offline

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Bringing online
- Rename (moving)
- Recovery

Managing disk space used by datafiles is made easier by enabling AUTOEXTEND which allows the database to enlarge a datafile when the tablespace runs out of free space. This attribute applies to the individual datafiles and not the tablespace. To resize the datafile manually, use the ALTER DATABASE DATAFILE statement with the AUTOEXTENT clause. The size extension limit and maximum limit must be assigned.

To take a datafile offline, use the ALTER DATABASE DATAFILE statement with the OFFLINE clause.

To bring a datafile online, use the ALTER DATABASE DATAFILE statement with the ONLINE clause.

To change the filename or move a datafile, use the ALTER DATABASE RENAME FILE statement.

To copy a datafile, use the HOST COPY statement.

To sync the file header with the database, use the RECOVER DATAFILE statement.

Schema Objects

A schema is a collection of database objects owned by a specific database user. The schema will take on the same name as the database user, making the two terms synonymous.

Schema objects include segments seen in tablespaces and nonsegment database objects owned by the user which include:

- Constraints
- Views
- Synonyms
- Procedures
- Packages

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Database objects not owned by the user and are not schema objects include:

- Roles
- Tablespaces
- Directories

Datatypes

Several datatypes are available for use in Oracle 10g database tables. They all fall within six major categories:

- Character
- Numeric
- Datatime
- Large Object (LOB)
- ROWID
- Binary

Character datatypes will store alphanumeric data in the database character set or the Unicode character set. The specified character set will indicate which language can be represented in the database. Here are the different character sets:

- US7ASCII supports English and all languages using a subset of the English alphabet.
- WE815O8859PI supports several European languages, including English, French, German, and Spanish.
- ALI6UTF16 is a Unicode character set and supports every known language, with some limits.

The database character datatypes are:

- CHAR(size [byte|char]), NCHAR(size) – fixed width types that always store column-width amount of data with padding to the right when needed. The NCHAR variation uses the Unicode character set.
- VARCHAR(size [byte|char]), VARCHAR2(size [byte|char]), NVARCHAR2(size) – variable width types that store only the amount of actually used data. The NVARCHAR2 variation uses the Unicode

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

character set. VARCHAR and VARCHAR2 are synonymous with Oracle 10g but may have different comparison semantics in future releases: the VARCHAR2 type is preferred.
- LONG a legacy datatype used for backward compatibility that stores variable-length alphanumeric data up to 2GB in size.

Numeric datatypes can store positive and negative fixed and floating-point numbers, zero, infinity, and the special 'Not A Number.' The database numeric datatypes are:
- NUMBER[(precision[,scale])] – stores zero, positive numbers, and negative numbers. Precision is the total number of digits and defaults to the maximum of 38. Scale is the number of digits to the right of the decimal point and defaults to 0.
- BINARY_FLOAT, BINARY_DOUBLE – store single-precision and double-precision floating-point data or a special floating-point values.

Datetime datatypes store dates, times, and time periods:
- DATE – stores a data time with a one-second granularity. The time portion will default to midnight.
- TIMESTAMP[(precision)] – stores a data and time with subsecond granularity. Precision is the digits of subsecond granularity ranging from 0 to 9 and defaulting to 6.
- TIMESTAMP[(precision)] WITH TIMEZONE – extends the TIMESTAMP datatype by storing a time zone offset which defines the difference (in hours and minutes) from the local time zone and UTC.
- TIMESTAMP[(precision)] WITH LOCAL TIMEZONE – extends the TIMESTAMP datatype by storing a time zone offset but converts the local time to the database time zone and backwards.
- INTERVAL DAY[(precision)] TO MONTH – stores a period of time in years and months. Precision is the maximum number of digits required for the year portion of the period with a range of 0 to 9 and a default of 2.
- INTERVAL DAY[(d_precision)] TO

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

SECOND[(s_precision)] – stores a period of time in days, hours, minutes, and seconds. D_precision is the maximum number of digits required for the day portion of the period ranging from 0 to 9 and a default of 2. S_precision is the maximum number of digits to the right of the decimal point needed for fractional seconds portion of this period, ranging from 0 to 9 and a default of 6.

- LOB datatypes store large objects up to 2^{32}-1 data blocks (4,294,967,295). LOBS are designed for text, image video, audio, and spatial data. When creating a table with LOB columns, a different tablespace and different attributes for the LOB data is specified separate from the rest of the table. A LOB locator acts as a pointer and is stored inline with the row to access the LOB data. The following LOB datatypes are supported:
- CLOB – variable-length character data.
- NCLOB – variable-length character data using the Unicode character set.
- BLOB – binary variable-length data inside the database.
- BFILE – binary variable-length data outside the database.

ROWIDs are physical or logical addresses that uniquely identify each row in the Oracle 10g table. The ROWID datatypes are:

- ROWID – stores the base64-encoded physical address of row in a heap-organized table in the database. ROWIDs incorporate the Object ID (OID), relative file number, block number, and row slot within the block.
- UROWID (Universal) – stores the base64-encoded physical address of row in an index-organized table.

Binary datatypes are used to store unstructured data and do not undergo character set conversion when passing from database to database through a database link, export/import utility, or between database client and server processes. The database binary datatypes are:

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- RAW(size) – stores unstructured data up to 2000 bytes in size.
- LONG RAW – stores unstructured data up to 2GB in size and exists for backward compatibility.

Tables

The primary data storage containers in Oracle 10g are tables. Data is organized in tables using rows and columns. When creating a table, it must have a name and the column names and datatypes must be specified. Table and column names have the following requirements:

- Must be from 1 to 30 bytes in length.
- Must begin with a letter.
- Can include letters, numbers, and special characters _, #, $.
- Cannot be a reserved word (NUMBER or INDEX).

If the name is enclosed in double quotation marks (“ “),the length of the name requires to be 1 to 30 bytes long and not contain an embedded double quotation mark. Each column name must be unique within the table. Each table name must be unique within the namespace. The name is a domain of allowable names for the set of schema objects and supports:

- Tables
- Views
- Sequences
- Private synonyms
- procedures
- Functions
- Packages
- Materialized views
- User-defined types

Databases also have separate namespaces for:

- Indexes
- Constraints
- Clusters
- Database triggers

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Private database links
- Dimensions
- Roles
- Public Synonyms
- Public database links
- Tablespaces
- Profiles
- Parameter files (PFILES)

Creating Tables

Use the CREATE TABLE statement to create a table. At minimum, the column names and datatypes must be listed. Commas delimit the column definitions starting with the column name. Some attributes can be added to the table definition. After creating the table, the structure of the table can be displayed using the SQL*Plus command, DESCRIBE.

A table can be created based on a query. Column attributes do not need to be specified as they are inherited from the existing schema object. Queries used in a CREATE TABLE AS SELECT (CTAS) statement can be a single table, view, or join multiple tables. Several options are also available:

- NOLOGGING – allows a database to log the contents of the table or subsequent direct path insert operations to the redo log.
- COMPRESS – allows the database to add data to the table used data compression.
- TABLESPACE – provides instructions about where to store the data.

A temporary table can be created which holds contents that are transitory and visible only to the session inserting data into it. The command statement is CREATE GLOBAL TEMPORARY TABLE. Though the definition of the table persists, the data in the table lasts for as long as the transaction or session lasts. To set how long the data should be kept, use the following options:

- ON COMMIT DELETE ROWS (transaction)
- ON COMMIT PRESERVE ROWS (session)

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Table Management

Default values for columns can be set when subsequent INSERT statements do not explicitly populate these columns. Having a default value does not ensure that a column will always have a value. An INSERT or UPDATE statement can explicitly set a column to NULL unless a NOT NULL constraint on the column. Default values can be any SQL expression not referring to a PL/SQL function, columns, or the pseudocolumns ROWNUM, NEXTVAL, CURRVAL, LEVEL, or PRIOR. More complex rules can be implemented using the BEFORE INSERT triggers. When specifying columns, default values can be defined when altering tables.

Descriptive comments can be added to the tables and columns to explain the content or usage of the database objects. Comments can be a maximum of 4,000 bytes in length and contain embedded white space and punctuation. To assign a comment, use the COMMENT ON statement and identify TABLE or COLUMN.

Tables, or other objects sharing a namespace, can be renamed using the RENAME statement. When renaming a table, the database will invalidate all objects that depend on the table. The database will automatically alter associated indexes, grants, and constraints to reference the new name.

The ALTER TABLE statement is another method of renaming the table. The statement can be used to add columns to the table. A single column requires an ADD option. Multiple columns can use the same option with the column name and specifications delimited by columns and the whole set being enclosed in parenthesis. To drop a column from a table, the DROP option is used. When dropping multiple columns, commas and parenthesis are required, but the specifications are not.

Columns can be modified using the ALTER TABLE MODIFY statement. The changes that can be made to columns are:

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Size of the column
- Column name
- Default value

Multiple columns are modified using commas and enclosing parenthesis.

The attributes of a table can be viewed using the DESCRIBE command from SQL*Plus or iSQL*Plus. This command will display the column names, datatypes, and nullity of the column. The physical attributes of the table can be viewed by querying the ALL_TABLES or USER_TABLES dictionary view. Constraints on the table can be seen by using the ALL_CONSTRAINTS and ALL_CONS_COLUMNS views.

Constraints

Business rules are enforced in the database using constraints, by limiting the acceptable data values for the table. Constraints are optional schema objects which are dependent on tables. A table can exist without a constraint, but a constraint cannot exist without a table. Several types of constraints can be created:

- NOT NULL
- UNIQUE
- PRIMARY KEY
- REFERENTIAL
- CHECK

Constraints can be added when creating a table or after its creation. They can be specified with either in-line syntax as a column attribute or out-of-line syntax as part of the table definition. Though constraints do not require a name, if one is not specified then Oracle will generate a name as a number with the prefix SYS_C which is not meaningful.

By default, all columns in a table allow NULL to be a valid value which represents unknown or nonexistent information. The NOT NULL constraint can be used to enforce business rules where information in a column must be known. NOT NULL constraints must be declared together with the column definition using in-line

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

syntax.

UNIQUE constraints ensure that each occurrence of the column protected by this constraint is different from all other occurrences on the table. These constraints cannot be created on the following column type:

- CLOB
- NCLOB
- BLOB
- LONG
- LONG RAW
- TIMESTAMP WITH TIMEZONE

The UNIQUE constraint can utilize an out-of-line or in-line syntax. As a result no two rows can have the same value, except NULL values. An index is created by the database to enforce this constraint.

The PRIMARY KEY constraint identifies any unique identifier for any tupple in an entity and enforce the validity of the primary key. A table can have only one PRIMARY KEY constraint. It includes both a NOT NULL and UNIQUE constraint attributes and limitations. This type of constraint can be created when creating a table.

The FOREIGN KEY constraints are also referred to as referential integrity constraints because they ensure data values referenced in one table are defined in another table. As a result, parent/child or referenced/dependent relationships are enforced by the constraints. When declaring a FOREIGN KEY constraint, the columns of one table must have values that appear as the primary key or unique key of another table. The table with the primary or unique key is considered the parent or reference table while the table with the FOREIGN KEY is considered the child or dependent table. A parent and child tables do not need to be separate tables, but can be separate columns of the same table.

CHECK Constraints verify that a column or set of columns meet a specific condition that must evaluate to a Boolean. If the condition evaluates to FALSE the database raises an exception and the INSERT or UPDATE statement fails. The condition

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

cannot include subqueries, references to other tables, or calls to nondeterministic functions.

Once created, constraints can be dropped, disabled, enabled, or renamed, using the ALTER TABLE statement.

Indexes

Indexes are data structures built on tables. They are options and improve data retrieval performance by providing a direct access method. An index key is one data value stored in the index. A Btree index sorts keys into a binary tree and stores these keys with the table's ROWIDs. In a bitmap index, a bitmap is created for each key. A bit exists in each bitmap for every ROWID in the table.

Btree indexes are the default index type. They can be unique, or non-unique. They can be used for medium- to high-cardinality columns. Btree indexes support row-level locking and appropriate for multiuser, transactional applications. Indexes supporting PRIMARY KEY or UNIQUE constraints are Btree indexes.

For multiple combinations of low- to medium-cardinality columns, bitmap indexes are appropriate. These indexes do not support row-level locking and are best for environments where changes to data are limited and controlled. Since bitmap indexes do not efficiently make changes to the indexed data, the indexes are dropped before data loading and recreated afterwards.

Managing Indexes

A table is required to create an index. When using the CREATE INDEX statement, the name of the new index, table they create the index on, and the columns to include must be identified. Multiple columns should be identified using a comma-delimited list.

The CREATE INDEX statement will create a Btree index. The

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

unique index requires an additional keyword for a complete statement, CREATE UNIQUE INDEX. A bitmap index also has a different keyword for a statement, CREATE BITMAP INDEX.

To drop as index, use the DROP UINDEX statement.

An index can be rebuilt, moved to a new tablespace, coalesced, or renamed. These actions can be done using an ALTER INDEX statement with several different clauses.

To rebuild an index, use a REBUILD clause. This action will typically shrink the size the index and possibly reduce the Btree depth which will make it more efficient.

To move an index from one tablespace to another, specify a new tablespace in conjunction with REBUILD TABLESPACE.

Coalescing is a different form of rebuild that combines entries in nearby leak blocks instead of re-creating the Btree. The intent of this process is to free space from some of the leaf blocks. The space and resources required for a coalesce is less than a full rebuild. Use the COALESCE clause.

Renaming an index can be performed using the RENAME clause.

Views

Views are created with a query incorporating one or more base tables. They become virtual tables used to select data, though some can be updated, deleted, or inserted into. Views can be used to restrict access to a subset of a table or simplify a complex table.

To create a view, use the statement CREATE OR REPLACE VIEW. The OR REPLACE segment of the statement requests that the database replace the view definition if one already exists. IF not included the statement may fail.

To remove a view from the database, sue the DROP VIEW

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

statement.

Sequences

Sequences are schema objects which generate unique integers, frequently primary key values. These objects are created using the CREATE SEQUENCE statement. At minimum , a name must be specified. The following attributes can be added to the statement:

- START WITH – defines the initial value.
- INCREMENY BY – defines the subsequent sequence numbers with the default being 1 and valid values being nonzero integers with less than 29 digits. Both negative and positive sequences are supported.
- MAXVALUE – defines the highest value that the sequence can generate with the default being a special keyword NOMAXVALUE, evaluating to 10^{27} for an ascending sequence and -1 for a descending sequence.
- MINVALUE – defines the lowest value that the sequence can generate with the default being a special keyword NOMINVALUE, evaluating to 10^{26} for an ascending sequence and -1 for a descending sequence.
- CACHE – defines how many values will be preallocated and held in memory with the default being 20 and the valid values ranging from 2 to the number returned from the formula:
  (CEIL (MAXVALUE – MINVALUE)) / ABS(INCREMENT BY)

These attributes can be changed using the ALTER SEQUENCE statement, except for the START WITH attribute.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Recovery Manager Features

The Oracle Recovery Manager (RMAN) is a backup and recovery tool provided with the Oracle Database Server 10g software. Several capabilities are in place to support and facilitate the backup and recovery process. The tool has a web-based GUI and command-line version.

The RMAN tool was first introduced with Oracle 8 and has been improved with each subsequent version. The latest version can handle larger databases backups and recoveries faster. Here is a list of RMAN features existing and new:

- Backup databases, tablespaces, datafiles, control files, and archive logs.
- Performing change-aware incremental backups.
- Providing scripting capabilities to combine tasks.
- Logging backup operations.
- Integration with third-party tape media software.
- Stores information about backups in a catalog within an Oracle database.
- Provides reports and lists of catalog information.
- Performance benefits.
- Duplicate database creation for testing and development.
- Backups testing to confirm successful restoration.
- Determining if backups are available in media libraries.
- Migration of datafiles across operating system platforms.
- User error recovery with flashback.
- Automated tablespace point-in-time recovery (TSPITR).
- Database dropping.
- Use of backup copies and flash recovery.
- Creation and use of RMAN backup copies.
- Configuration of default disk backup types.
- Recovery of datafiles not backed up.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Blocking change tracking.
- Un-registration of backups in catalog.
- Actual compression of RMAN backups.
- Error-reporting improvements.

RMAN Components

Several components of the RMAN allow the backup and recovery process to be standardized and automated.

Access to the Recovery Manager is available through a web-enabled GUI or command line interface (CLI). The process utilizes off-server sessions that connect to the target database to be backed up. GUI access is available through the Oracle Enterprise Manager's web-based console. The EM tool performs backups, imports, exports, data loads, performance monitoring and tuning, lob and event scheduling, and standard DBA management.

The recovery catalog is a set of recovery information contained within an Oracle database. It is similar to the RMAN repository stored in the control file but contains more extensive information. Ultimately, it is a data dictionary storing backup information in a set of tables. This catalog provides information about backups, restores, and recoveries, including the status of success or failure of several types of backups including:

- Operating system backups
- Datafile backups
- Tablespace copies
- Control file copies
- Archive log copies
- Full database backups
- Physical database structures

RMAN commands allow various actions to be performed to facilitate the backup and restoration of a database. They are organized into scripts in a logical order and stored in the recovery catalog database. The scripts are reused for other backups, allowing consistency throughout different target

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

database backups.

Tape media connectivity enables an interface with various third-party tape hardware vendors to store and track backups in automated tape libraries (ATLs). Support is available for several tape hardware devices and ATLs are tape units that use robotics arms and bar-coded tapes to automate the usage of multiple tapes for backup purposes.

RMAN Usage

RMAN backup is a physical backup method. To use it, a disk or tape must be used. When using tape, a third-party media management library is required to interface with the tape hardware.

Two other methods for backup and recovery of an Oracle database are available: user-managed and the Oracle EXPORT utility. User-managed backups are customized scripts which interact with operating system capabilities such as copy and compress commands.

The EXPORT utility using the exp executable is a logical backup utility that performs backups only on the logical components of the database. A physical backup consists of backing up the database files like datafiles, control files, and redo logs. A logical backup consists of backing up at the tablespace level or other logical database components. The utility is typically considered additional backup protection for the RMAN or user-managed backups rather than a stand-alone backup method.

RMAN provides recovery options that are not available in other backup and recovery methods. The one major backup capability not supported by RMAN is the backup of password files, which can be done through user-managed backups. The other backup capabilities that are supported include:

- Server parameter files
- Closed database
- Open database
- Incremental

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Corrupt block detection
- Automatic backup file
- Backup catalogs
- Media manager
- Platform independent

Repository and Control Files

Metadata about target databases are collected. The entire collection is called the RMAN repository. Two methods of storing information about the target databases are available through the RMAN utility. The first method accesses an RMAN catalog of information about backups. The second method accesses the necessary information about backups in the target database's control files. For medium-sized to enterprise database environments, Oracle recommends the use of a catalog database over the RMAN repository to enable fill functionality of the RMAN utility.

If RAMN is used without the recovery catalog, most information about the target database is stored in the target database's control file. In this situation, the control file must be managed appropriately. The parameter CONTROL_FILE_RECORD_KEEP_TIME in init.ora or spfile.ora specifies how long the backup information used by RMAN is kept in the control file. The default value is 7 days but can be configured for up to 365 days. The longer the information is kept, the larger the control file becomes.

The information stored in the control file is stored in reusable section called circular reuse records and non-circular reuse records. The circular reuse records have non-critical information that is overwritten when required. Datafiles and redo log information can be contained within the non-circular reusable records.

Recovery Catalog

The recovery catalog is designed to be a central storage place

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

for multiple databases' RMAN information. Instead of managing distinct control files for multiple target databases in separate locations, the Recovery Catalog allows centralization of the RMAN information.

The main components of the RMAN recovery catalog support the logging of backup and recovery information. This information is stored in tables, views, and other database objects. These components include:

- Backup and recovery information
- RMAN scripts
- Backup information on datafiles and archive logs
- Information on physical makeup or target database

The recovery catalog must be stored on its own database, preferably separately on a server separate from the server where the target database resides. To enable the catalog, an account must be created to hold the catalog tables with the following privileges:

- CONNECT
- RESOURCE
- RECOVERY_CATALOG_OWNER

In addition to the account, a catalog creation script command must be executed as the user RMAN_USER connected to the RMAN utility.

Creating a Recovery Catalog takes on the following steps:

1. Point to the database where the recovery catalog is to reside.
2. Create the user that will store the catalog.
3. Grant the appropriate permissions to the RWAN user.
4. Launch the RMAN tool.
5. Connect the catalog with the created user.
6. Create the recovery catalog and specify the residing tablespace with the following command in RMAN:
   CREATE CATALOG TABLESPACE DATA

Once a target database has been registered, it can be backed up. The backup data will be stored in the recovery catalog. To perform a backup, the target database must be connected, as

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

well as the recovery catalog. The appropriate RMAN backup script must be executed. The entire database will be backed up. Another RMAN script can be used to recover the database.

Starting and Connecting to RMAN

RMAN can be started by launching the RMAN executable. It can be stopped by exiting or quitting the RMAN prompt. After accessing the RMAN utility, backup configuration changes and restores can be performed. There are three database connection types using the RMAN utility:

- Target database
- Recovery catalog database
- Auxiliary database

The auxiliary database is a standby database, duplicate database, or auxiliary instance.

Connecting to Target Database

The target database can be connected to using two methods: the command line and the RMAN utility.

To connect to the target database using the command line, follow these steps:

1. Set the Oracle system identifier (SID).
2. Launch the RMAN executable, specifying the target and default connection in the command line.

To connect to the target database using the RMAN utility, similar steps are performed from the RMAN command prompt rather than the operating system's command line.

1. From the operating system, set the Oracle SID.
2. Launch the RMAN executable.
3. Connect to the target database using the CONNECT TARGET command.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Connecting to Recovery Catalog

Connecting to a recovery catalog can also be performed through the command line or the RMAN utility. The steps for connecting through the command line are:

1. Set the Oracle SID.
2. Launch the RMAN executable, specifying the target, the catalog and catalog owner, and the database containing the catalog.

To connect to the recovery database from the RMAN utility, follow these steps:

1. From the operating system, set the Oracle SID.
2. Launch the RMAN executable.
3. Connect to the target database using the CONNECT TARGET command.
4. Connect to the recovery catalog, specifying the username and password of the RMAN catalog owner.

Configuring RMAN

Media Management Layer

The media management layer (MML) is an Application Programming Interface (API) which interfaces with RMAN and several hardware vendors' tape devices, or ATLs.

All tape hardware vendors designed to work with Oracle RMAN make their own MML. Most hardware devices are proprietary and require different program calls. The MML ensures communication with the Oracle database kernel to allow the RMAN server process and the MML to read and write Oracle data to the physical tape drive.

Channel Allocation

Channel Allocation is a method of connecting RMAN and the

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

target databases while determining the type of I/O devices used by the server process to perform the backup or restore operation. Channels are allocated either manually or automatically.

Manual channel allocation is performed any time an ALLOCATE CHANNEL command is issued. The syntax of the command to write to a disk file system is 'ALLOCATE CHANNEL channel name TYPE DISK.' The syntax of the command to write to a tape backup system is 'ALLOCATE CHANNEL channel name TYPE tape name.'

Automatic channel allocation is performed by setting the RMAN configuration at the RMAN command prompt. Using the commands CONFIGURE DEFAULT DEVICE or CONFIGURE DEVICE command swill set the appropriate configurations. This method is automatically used when the BACKUP, RESTORE, or DELETE commands are executed.

Channel control options or commands are used for either manual or automatic allocation of channels. They control the operating system resources that RMAN uses when performing RMAN operations. Here a list of options or commands:

- ALLOCATE CHANNEL / CONFIGURE CHANNEL RATE – limits the I/O bandwidth in KB, MB, GB.
- ALLOCATE CHANNEL / CONFIGURE CHANNEL MAXPIECESIZE – limits the size of the backup pieces.
- ALLOCATE CHANNEL / CONFIGURE CHANNEL MAXSETSIZE – limits the size of the backup sets.
- SEND – sends vendor specific commands to the Media Manager.
- ALLOCATE CHANNEL / CONFIGURE CHANNEL CONNECT – instructs specific instances to perform an operation.
- ALLOCATE CHANNEL / CONFIGURE CHANNEL PARMS – sends vendor specific parameters to the Media Manager.

Automatic channel failover requires multiple channels to be allocated. If a failure occurs in the backup or restore operations,

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

RMAN will complete the operations using other available channels.

RMAN Parameters and Persistent Settings

Configuration of persistent settings or parameters for RMAN is done through the configuration settings for each target database. These settings are used to simplify or automate the use of RMAN. The settings are stored in the control files of the target databases or the recovery catalog.

To display the configuration of a particular RMAN session:

1. Set the Oracle SID.
2. Launch the RMAN utility.
3. Connect to the target database.
4. Perform the SHOW ALL command to display all configuration parameters.

Some commonly used configuration settings for using RMAN include:

- DEVICE TYPE
- BACKUP TYPE
- COMPRESSED BACKUPSET
- CHANNEL DISK DEVICE
- CHANNEL TAPE DEVICE

EM Configuration of RMAN Settings

The EM is a web-based console with a home page utilizing the following syntax:

http://hostname.domain:EM portnumber/em

The hostname is the server and the domain is the network. The default port number and the one used to for the first database running EM is 5500. The port number for each additional database is incremented by 1.

To configure RMAN settings through the EM console:

1. Enter the URL for the EM home page and log in.
2. Access the Maintenance table.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

3. Select the Backup/Recovery section and click Configure Backup Settings.
4. Set or modify the displayed backup, recovery, recovery catalog configuration settings.

Retention Policies

A retention policy describes the length of time a backup is retained for use by a possible restore. It is determined by the configuration parameter RETENTION POLICY and can be displayed with the SHOW ALL command.

Backups can be modified to block or obsolete their status from the retention policy. To block a backup from the retention policy, use the commands CHANGE and KEEP. To obsolete the backup from the retention policy, use the commands CHANGE and NOKEEP.

Control File Autobackup

To automatically back up the control file whenever information impacting the control file is changed or modified can be done through the RMAN. This is an advantage to a backup because it allows a database to be recovered even if the control file or server parameter file is lost. The process is called control file autobackup.

The common naming of the server parameter file and control file allows RMAN to search and restore the files without accessing the RMNA repository. After the control file is restored and mounted, the EMAN repository becomes available and its information can be used to restore datafiles and archive logs.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

RMAN Commands

Here are some common commands and clauses for RMAN:

- @ - run a command file.
- @@ - run a command file in the same directory as another command file currently running.
- ALLOCATE CHANNEL – establish a connection between the RMAN and database instance.
- ALLOCATE DHANNLE FOR MAINTENANCE – establish a connection to prepare for maintenance commands.
- allocOperandList – subclause that specifies channel control options.
- ALTER DATABASE – mount or open database.
- ArchivelogRecordSpecifier – specify a range of archived redo log files.
- BACKUP – back up database files, copies of database files, archived logs, and backup sets.
- BLOCKRECOVER – recover individual data blocks or sets of data blocks within one or more datafiles.
- CATALOG – add information about datafile copy, archived redo log, or control file copy to the repository.
- CHANGE – mark a backup piece, remove a repository record, override the retention policy.
- completedTimeSpec - specify a time range during which the backup or copy completes.
- CONFIGURE – configure persistent RMAN settings.
- CONNECT – establishes a connection between RMAN and a target, auxiliary, or recovery catalog database.
- connectStringSpec – specify the username, password, and net service name for connecting to a target, recovery catalog, or auxiliary database.
- CONVERT – converts datafile formats fro transporting tablespaces across platforms.
- CREATE CATALOG – create schema for the

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

recovery catalog.
- CREATE SCRIPT – create stored script and store in the recovery catalog.
- CROSSCHECK – determine whether files management by RMAN still exist on disk or tape.
- datafileSpec – specify a datafile by filename of absolute file number.
- DELETE – delete backups and copies, remove references and update control files records.
- DELETE SCRIPT – delete a stored script from the recovery catalog.
- deviceSpecifier – specify the type of storage device for backup or copy.
- DROP CATALOG – remove a schema from the recovery catalog.
- DROP DATABASE -delete the target database from disk and unregister it.
- DUPLICATE – use backups of the target database to create a duplicate database.
- EXECUTE SCRIPT – run an RMAN stored script.
- EXIT – quit the RMAN executable.
- fileNameConversionSpec – specify patterns to transform source to target filenames.
- FLASHBACK – return the database to its state at previous time of system change number (SCN).
- formatSpec – specify a filename format for a backup or copy.
- HOST – invoke an operating system command-line subshell from within RMAN.
- keepOption – specify that a backup or copy should or should not be exempt from the current retention policy.
- LIST – produces a detailed listing of backup sets or copies.
- listObjList – a subclause used to specify which items will be displayed by the LIST command.
- maintQualifier – a subclause specifying additional options for maintenance commands.
- obsOperandList – a subclause determining which backups and copies are obsolete.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- PRINT SCRIPT – displays a stored script.
- QUIT – exits the RMAN executable.
- RecordSpec – subclause specifying which objects the maintenance commands should operate on.
- RECOVER – apply redo logs and incremental backups to datafiles.
- REGISTER – registers the target database in the recovery catalog.
- RELEASE CHANNEL – release channel that was allocated with the ALLOCATE CHANNEL command.
- releaseForMaint – releases channel that was allocate with an ALLOCATE CHANNEL FOR MAINTENANCE command.
- REPLACE SCRIPT – replace existing script stored in the recovery catalog.
- REPORT – perform detailed analysis of the content of the recovery catalog.
- RESET DATABASE – informs RMAN that the SQL statement ALTER DATABASE OPEN RESETLOGS has been executed.
- RESTORE – restore files from backup sets or from disk copies to the default or new location.
- RESYNC – performs a full resynchronization.
- RUN – executes a sequence of one or more RMAN commands.
- SEND – send a vendor-specific quoted string to one or more specific channels.
- SET – sets value of various attributes that affect RMAN behavior for the duration of the RUN block or session.
- SHOW – displays the current CONFIGURE settings.
- SHUTDOWN – shut down the target database.
- SPOOL – write RMAN output to a log file.
- SQL – execute a SQL statement from within RMAN.
- STARTUP – starts up the target database.
- SWITCH – specifies the datafile that is pointed to by the control file.
- UNREGISTER DATABASE – unregister a database from the recovery catalog.
- untilClause – subclause specifying an upper limit by

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

time, SCN, or log sequence number.
- UPGRADE CATALOG – upgrades the recovery catalog schema from and older version.
- VALIDATE – examines a backup set and report whether its data is intact.

Commands can be grouped together within a RUN block.

Backup Sets and Image Copies

A backup set is a logical grouping of backup pieces stored in a proprietary format. Backup pieces are actual files within the backup set. Database files in backup sets are stored in a special RMAN format and processed with the RESTORE command to allow the files to be usable. To perform the backup process on a backup set, use the RMAN BACKUP command.

A database that is backups is stored in two backup sets:
- FULL DATAFILE BACKUPSET
- ARCHIVE LOG BACKUPSET

The first backup set consists of two backup pieces, while the other backup set consists of a single backup piece. All backup pieces are stored in the System32 directory by default. The first backup piece contains the majority of the datafiles.

Backup sets allow multiplexing, enabling multiple files to be read and each file block being written to the same backup set. These blocks are interspersed together. Image files cannot be multiplexed.

Image copies are actual copies of the database files, archive files and control files. They are not stored in a special RMAN format and they can only be stored on disk.

No RMAN restore process is required to make image copies usable in a recovery event, improving the speed and efficiency of the restore and recovery process. Image copy backups cannot be compressed, thus requiring greater space than a backup set. The RMAN COPY command can be used to create an image

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

copy.

Backups of image copies can be simplified using the BACKUP AS COPY command which allows an image copy to be made of an entire database, multiple tablespaces, datafiles, and archive logs without specifying the individual files.

Compressed Backups

Backups in Oracle 10g can be compressed regardless of the contents of the datafiles. Compressed backups only work with backup sets. To create a compressed backup, use the BACKUP AS COMPRESSED command. The process proceeds just like a normal backup set, but it is compressed.

A default device can be configured for compressed backups using the CONFIGURE command. When executed, all backups on a designated device will be compressed until the configuration is changed.

Full and Incremental Backups

A full backup will back up all the data blocks in the datafiles, whether they are modified or not. An incremental backup will only back up the data blocks in the datafiles that were modified since the last incremental backup. A full backup cannot be part of an incremental backup strategy. The baseline backup for an incremental backup is designated as level 0. This level 0 backup is a full backup since all blocks are backed up regardless of modification. Incremental backups are than merged with the level 0 backup in the future to complete a full backup at the current point of time.

Two types of incremental backups exist:

- Differential
- Cumulative

A differential incremental backup will back up only data blocks modified since the more recent backup at the same level of

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

lower. An occurrence of the level 1 or level 2 backup is determined and any modifications made since the last backup is included. This is the default method for incremental backups.

A cumulative incremental backup will back up the data blocks that have been modified since the most recent backup of the next lowest level, or n-1. As a result, only one cumulative incremental backup needs to be restored rather than multiple differential incremental backups. A cumulative incremental backup requires more space than a differential incremental backup.

Parallelization

Backups on multiple backup sets can be performed concurrently over multiple device channels. This is called parallelization. It is done by allocating multiple channels, one for each backup set. The allocation can be done before the back up process begins. To perform parallelization, either modify the CONFIGURE settings for channel parallelism greater than 1 or manually allocate channel.

Backup Options

RMAN provides options for controlling filenames, backup performance, and the size of the backups. Here is a list of commands for options:

- FORMAT – control filenames.
- TAG – control filenames used for image copies and backup sets.
- RATE – limits I/O bandwidth usage.
- DURATION – set maximum time for backup to process.
- MAXSETSIZE – controls size of backup set.
- MAXPIECESIZE – controls size of backup piece

The FORMAT command has several variables that can be applied. Here are a few of those variables:

- %a specifies the activation ID of the database.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- %c specifies the copy number of the backup piece.
- %d specifies the database name.
- %D specifies the current day of the month.
- %e specifies the archived log sequence number.
- %f specifies the absolute file number.
- %F combines the DBID, day, month, year, and sequence into a unique and repeatable generated name.
- %h specifies the archived redo log thread number.
- %I specifies the database ID.
- %M specifies the month in the Gregorian calendar.
- %N specifies the tablespace name.
- %n specifies the database name, padded to the right to a total length of 8 characters.
- %p specifies the piece number within the backup set.
- %s specifies the backup set number.
- %t specifies the backup set timestamp.
- %T specifies the year, month, and day.
- %u specifies the 8 character name created by representation of backup set or image copy number.
- %U specifies the system-generated unique filename.

Block Change Tracking

A new feature of Oracle 10g is block change tracking. The process records all blocks modified since the last backup and stores then in a block change tracking file. The file is used by RMAN to determine the blocks that were backed up in an incremental backup. Since RMAN can access this file for the information instead of scanning the whole datafile, backup performance is improved.

A SQL command is used to enable and disable block change

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

tracking. By default, the process is disabled. To enable the process:
Use the ALTER DATABASE statement with ENABLE BLOCK CHANGE TRACKING. A location has to be designated for the file. To disable the process, use ALTER DATABASE DISABLE BLOCK CHANGE TRACKING.

Monitoring Backups

Actual sessions of RMAN backups and recoveries can be monitored using RMAN dynamic views. These views provide more information than is available thorough general RMAN reporting features of the LIST, REPORT, and SHOW commands. These dynamic views are accessed through SQL*Plus.

The following views are stored in the target database:

- V$RMAN_OUTPUT – displays messages reported by an RMAN job in progress.
- V$RMAN_STATUS – displays the success or failure of all completed RMAN jobs.
- V$PROCESS – shows currently active processes.
- V$RECOVER_FILE – shows datafiles that require recovery.
- V$SESSION – shows active sessions.
- V$SESSION_LONGOPS – shows progress reports on RMAN backups and restores.
- V$SESSION_WAIT – shows events causing sessions to wait.
- V$BACKUP_SYNC_IO – shows the rows when the I/O is synchronous to backup process.
- V$BACKUP_ASYNC_IO - shows the rows when the I/O is asynchronous to backup process.

LIST Commands

The RMAN repository can be queried using LIST commands. These commands can also get data regarding the BACKUP and COPY commands and database incarnations, which are a

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

separate version of a physical database. The output of the LIST command displays files where the CHANGE, CROSSCHECK, and DELETE commands have been used.

Several options are available for the LIST command:

- BY BACKUP
- BY FILE
- SUMMARY
- VERDOSE

REPORT Commands

The REPORT command allows the RMAN repository to be queried and get data with regard to files needing a backup, not needing a backup, database physical schema, and status of unrecoverable operations on files. The output of the REPORT command generates more detailed information from the RMAN repository.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Database Recovery

An Oracle 10g database can be recovered by:

- A traditional user-managed recovery procedure
- Recovery Manager (RMAN)
- Enterprise Manager (EM)

Recovery Process

Two phases of database recovery process exists: restoration and recovery.

The restore process focuses on copying required physical files required to begin the recovery phase. In a user-managed recovery, the restore process is completed manually and dependent on operating system commands or third-party tape management solution to retrieve the files from disk or tape. When using RMAN or EM, restores can be performed using the RESTORE command.

The recovery process will take a datafile and apply archived redo logs.

Server-Managed Recovery

The RMAN utility is the Oracle recommendation for performing both backup and recovery processes. The use of the RMAN utility for recovery is performed by a server process initiating the recovery process, commonly called server-managed recovery.

The process for server-managed recovery is:

1. DBA starts RMAN.
2. A server session is started.
3. A connection with the target database is established.
4. If the Recovery Catalog is not used, the control file is read as the repository.
5. The appropriate database files and archived redo logs are determined

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

6. Database files are restored and recovered.

The RESTORE command is to copy the datafiles, tablespaces, control files, archived redo logs, and server parameter files from disk or tape. This command is required for all backups created using the BACKUP command, because the backup is stored in a proprietary RMAN command format called a backup set. The RESTORE command is not required for the BACKUP AS COPY command.

The RECOVERY command is required for all backups made with the BACKUP or BACKUP AS COPY commands. The RECOVER command will apply the archived redo logs to the restored data files to apply the changes that are not present in the restored database files.

User-Managed Recovery

The traditional recovery method involves direct management of the database files to recovery the database. Referred to as user-managed recovery, this method involves the use of operating system command.

The basic procedure for user-managed recoveries is:

1. Identify the database files to be restored.
2. Place database in mounted or open mode.
3. Restore files from tape of disk to the appropriate location.
4. Use RECOVER command to alloy archived redo logs to the database files.

Recovering Control Files

Control files are key database files required for the recovery process because they contain RMAN metadata information and required recovery information. In Oracle 10g, a control file autobackup process is available to configure RMAN to backup the control file to the flash recovery area.

To configure a control file autobackup, configure the RMAN

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

setting including enabling a configuration parameter. Once set, the control file autobackup will configure all backups to perform an automatic backup on the control file. A RMAN backup should be performed after the entire configuration has been completed. The RMAN backup will contain a control file used for recovery operations. With this control file backup, a recover control file operation can be performed. If the recovery catalog is not used, the database identifier (DBID) should be specified after connecting to the target database.

Performing Incomplete Recoveries

A recovery that stops before the failure that forced the recovery is called an incomplete recovery, and results because not all transactions in the archived redo logs are applied to the database to make it complete. An incomplete recovery is also called a database point-in-time recovery (DBPITR).

To perform a RMAN incomplete recovery, use the following clauses with the RECOVER command:

- Use before the RECOVER command -
  - SET UNTIL TIME
  - SET UNTIL SEQUENCE
- With the RECOVER command in RMAN -
  - UNTIL TIME - stop recovery before a known point that would introduce an unrecoverable corruption and some undesired event.
  - UNTIL SEQUENCE – stop recovery before a known redo log sequence that would introduce an unrecoverable corruption and some undesired event.
  - UNTIL SCN – stop recovery before a known SCN that would introduce an unrecoverable corruption and some undesired event.
- With the RECOVER command in user-managed recovery
  - UNTIL TIME - stop recovery before a known point that would introduce an unrecoverable corruption and some undesired event.
  - UNTIL CHANGE - stop recovery before a

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

known SCN that would introduce an unrecoverable corruption and some undesired event.
    - UNTIL CANCEL – step when administrator issues the CANCEL command and typically used to create a test database.

Recovery Through Enterprise Manager

A database recovery using the Enterprise Manager follows the same process as performing a recovery through the RMAN command-line interface. The only difference is the use of the EM web interface. The process still uses the operating system commands to log on to the server and run the RMAN CLI.

To perform the database recovery using EM, the database must be shut down and started in MOUNT mode. The remaining steps are:

- Enter the EM main home page through a web browser.
- Select the Maintenance tab.
- Select Perform Recovery under the Backup/Recovery section.
- Select Whole Database from the Object Type list box.
- Enter the username and password with administrator privilege.
- Through the Recovery Wizard, the RMAN performs a shutdown of the database.
- From the Database: Ora101 screen, click the Startup button to start the database in MOUNT mode.
- Enter host credentials with administrator privileges.
- Choose Recover To A Prior Point in Time radio button and proceed.
- Choose to restore files to the default location.
- Review and submit the RMAN script to be executed.

RESETLOGS Operation

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

In most incomplete recovery situations, the RESETLOGS clause is required to open the database. In previous versions of Oracle database, this step was critical to allow databases to function properly. Though this is still true for Oracle 10g, the feature is not internal to the recovery process.

It is recommended that after an incomplete recovery, an immediate backup of the database should be performed.

Using the Flashback Database

The Oracle Flashback Database is a specific Flashback Technology within the Oracle 10g architecture. The remaining technologies include:

- Flashback Query
- Flashback Drop
- Flashback Versions Query
- Flashback Transaction Query
- Flashback Table

All Flashback Technologies, except the Flashback Database, are based on undo data. The Flashback Database is effective to replace an incomplete recovery of a complete database and is generally faster and more efficient. The Flashback Technologies, with the exception of the Flashback Database, are designed to support end users in repairing their own errors, reducing work requirements on the DBA. The Flashback Database still requires DBA support and allows recovery of errors like truncating a large table, incomplete batch jobs, or dropped user.

Flashback Databases

The Flashback Database technology is a new feature of Oracle 10g which builds on the flash query capabilities of the previous Oracle version. The feature allows the entire database to be flashed back to a specific point-in-time. Because their impact on the database is unknown, user errors and logical corruptions are the database problems are best resolved using the Flashback

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Database feature; though it is not effective in every situation, such as resizing of a database or a deleted datafile.

The Flashback Database is not impacted by the size of the database like traditional recovery options. The recovery time is dependent on the number of changes that need to back out of the recovery process. The architecture for the Flashback Database feature includes a recovery writer (RVWR) background process and Flashback Database logs. When enabled, the RVWR process starts writing Flashback Database logs in the flash recovery area. These logs contain an image of the physical database blocks in a "before" state. The flash recovery area is a prerequisite that must be enabled to allow logs to be written to it.

Though the Flashback Database is a good substitute for incomplete recovery, it does have its limitations:

- Cannot resolve media failures.
- Cannot resolve shrinking datafiles.
- Cannot be used if control files have been restored or recreated.
- Cannot drop a tablespace and recover though RESETLOGS.
- Cannot flash back to an SCN prior to the earliest available SCN in the flashback logs.

Managing the Flash Recovery Area

The flash recovery area is designed to centralize the storage of all recovery files in a specific location on the disk, which can be a local disk, a disk subsystem like a Storage Area Network (SAN), a Network Attached Storage (NAS), or Oracle Automatic Storage Management (ASM). Flash recovery provides some key benefits to the recovery process:

- All recovery files are stored in a centralized location.
- The method of backup and restoration is faster, because the information is written to disk rather than tape.

Initializing a database parameter will create a flash recovery and define the size and location of the flash recovery area. As files

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

are added or deleted from the flash recovery area, the information is recorded in the database alert logs. The DBA_OUTSTANDING_ALERTS is a database view that can provide information about the status of the flash recovery area. The RMAN retention policy will determine the retention period of the file in the flash recovery area in increments of days. Only files with retention policies can be deleted from the flash recovery area, such as files that exceed the retention policy or are obsolete.

Alerts can be sent to the alert log when the flash recovery area reaches 90% or 95% capacity. At 90% full, obsolete files will be automatically removed from the area.

Configuring the Flash Memory Area

To create a flash recovery area, add database parameters specifying the size and location of the area. If the flash recovery area already exists, the size of the flash recovery area can be changed or it can be disabled.

Using the Flash Memory Area

All recovery-related files are managed in the centralized flash recovery area. These files are related with the datafiles and control files, thus reducing the manual cleanup of unneeded archive logs. After setting up the flash recovery area, specify the area as the default for backups using the CONFIGURE command. The BACKUP or BACKUP AS COPY command can be performed.

Statistics such as space consumed can be viewed by querying V$RECOVERY_FILE_DEST. The dynamic view will display information about the usage of the flash recovery area. The RMAN should be accessed and back up the recovery area.

The flash recovery area can be used in other ways:

- To create Oracle database files.
- To contain datafiles, control files, online redo logs,

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

miscellaneous RMAN files, and flashback logs.

To take advantage of the flash recovery area fully, store redo logs, control files, and archived redo logs when they are created. The following commands are used to create online redo logs in the flash recovery area:

- CREATE DATABASE
- ALTER DATABASE ADD LOGFILE
- ALTER DATABASE ADD STANDBY LOGFILE
- ALTER DATABASE OPENRESET LOGS

The location of the online redo log files can be decided through the following initialization parameters:

- DB_CREATE_ONLINE_LOG_DEST_n
- DB_RECOVERY_FILE_DEST
- DB_CREATE_FILE_DEST

The location of the control files can be decided through the following initialization parameters:

- CONTROL_FILES
- DB_CREATE_ONLINE_LOG_DEST_n
- DB_RECOVERY_FILE_DEST
- DB_CREATE_FILE_DEST

Some behaviors should be considered when using Oracle Managed Files (OMF) which is a precursor to Automated Storage Management (ASM). The DB_RECOVERY_FILE_DEST and DB_CREATE_FILE_DEST initialization parameters used in conjunctions with the OMF, a control file will be created in the location of the parameter. When using DB_CREATE_ONLINE_LOG_DEST_n parameter with OMF, the control file is create in 'n' number of locations: the first location being the primary control file. When using the DB_RECOVERY_FILE_DEST parameter with OMF, the control file is placed in the flash recovery area.

The initialization parameters that impact the location of archived log files are:

- LOG_ARCHIVE_DEST_n
- LOG_ARCHIVE_DEST

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- LOG_ARCHIVE_DUPLEX_DEST

Only the LOG_ARCHIVE_DEST_n parameter can be used to create archived log files in the flash recovery area.

Backup the Flash Recovery Area

Since the flash recovery area contains important backup information critical to the recovery of the database, it is important to backup the flash recovery area. This is performed to a tape drive. The commands specially designed to backup the flash recovery area includes:

- BACKUP RECOVERY AREA will backup the whole recovery area.
- BACKUP RECOVERY FILES will backup only the files in the recovery area.

These commands must be used in conjunction with a tape device and a media manager configured to use with RMAN.

Configuring the Flashback Database

The database has several features that need to be configured prior to configuring the Flashback Database, specifically enabling the ARCHIVE LOG. To configure the Flashback Database, the database must be shutdown and started in MOUNT mode. The database parameter DB_FALSHBACK_RETENTION_TARGET can be set to the desired value, in minutes, to determine how far back in time the database should be flashed back. The Flashback Database can be enabled using the ALTER DATABASE FLASHBACK ON command. At this point, the database is now open for use.

Using RMAN with the Flashback Database

The RMAN can be used to perform recoveries with the Flashback Database. To perform this backup, mount the database that is configured for the Flashback Database. Then perform a Flashback Database Recovery. Additionally, either

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

OLDEST_FLASHBACK_SCN or OLDEST_FLASHBACK_TIME should be retrieved from the V$FLASHBACK_DATABASE_LOG view to allow the TO SCN and TO TIME clause to be utilized in the FLASHBACK DATABASE clause. The OT SEQUENCE clause uses the redo log sequence and thread to perform the recovery.

Monitoring the Flashback Database

Several views can be used to monitor the Flashback Database:
V$DATABASE
V$FLASHBACK_DATABASE_LOG
V$FLASHBACK_DATABASE_STAT

The V$DATABASE view will display whether the Flashback Database is on or off and will provide information about it being enabled or not. The V$FLASHBACK_DATABASE_LOG aids in determining the amount of space required in the recovery area to support the flashback activity generated by changes in the database.

Using EM with the Flashback Database

When using the Flashback Database with the Enterprise Manager, the same prerequisites and configuration of the flash recovery area required to use the feature with RMAN must be performed. This can be done manually through SQL or through the Maintenance and Configure Recovery Settings screens in EM. Once the configuration is complete, a recovery can be performed.

To configure the Flashback Database with EM, log in with a SYSDBA account. Go to the Maintenance tab and choose Configure Recovery Settings under the Backup/Recovery section. The first parameters to be configured are the location and size of the flash recovery area. Also configurable are the parameters to enable flashback logging and flashback retention target.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

A Flashback Recovery using the EM utility is performed just like a normal recovery. The EM will determine whether the recovery can be performed using the Flashback Database based on selections in the recovery screens.

Non-Critical Losses

Non-critical files are database files can be essential to the operations of a database, but not have a critical impact on those operations should they be compromised. Recovery of these files also has a minimal impact to database operations. These files include:

- Temporary tablespaces
- Redo log files
- Index tablespaces
- Indexes
- Read-only tablespaces
- Password files

Temporary Tablespaces

Database sorting operations are possible because of temporary tablespaces. These files are part of the physical database files which are expected to exist under normal operations. No permanent objects are stored in temporary tablespaces, and therefore, no change is made to the system change number (SCN) from the checkpoint process in the control file. Except for creating temporary segments, the database will function normally when a temporary tablespace is lost. Because only temporary data is stored within, the temporary tablespace can be re-created and reassigned after being lost. Recreation of the temporary tablespace can be done using the CREATE TEMPORARY TABLESPACE command.

A tempfile is a type of tablespace that is managed locally or in the tablespace, rather than in the data dictionary. They are commonly used with temporary tablespaces and are a default for the Database Configuration Assistant (DBCA) creation of

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

temporary tablespaces.

To start a database with a missing tempfile:

1. Start and mount the database as if not already running.
2. Drop the tablespace.
3. Re-create the temporary tablespace.

Another tablespace recovery method is by modifying the existing temporary tablespace to a new or different temporary tablespace, using the ALTER DATABASE DEFAULT TEMPORARY TABLESPACE command.

Redo Log Files

Redo logs contain all committed and uncommitted transactions and are an important element of the database physical structure. A standard for creating an Oracle database is to utilize mirrored redo logs, or multiplexed redo logs. These are multiple redo logs which are members of a redo log group. If a redo log member is lost or deleted and the mirrored log member still exists, the original member can be easily rebuilt. As long as one mirrored redo log is available, archive logs will still be processes until the switch to the next log file sequence.

To create a log file member, use the ALTER DATABASE ADD LOGFILE MEMBER statement.

Index Tablespaces

Index tablespaces contain indexes which are created from the underlying database tables. The process for recovering an index tablespace is:

1. Start and mount the database as if not already running.
2. Drop the tablespace.
3. Re-create the index tablespace using CREATE TABLESPACE INDEXES command.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Indexes

After recreating the index tablespace, the re-creation of the index is required. Any existing indexes will be lost when the tablespace is created from scratch. Many times, copies of the indexes may already be present to perform various maintenance activities.

Read-Only Tablespaces

Tablespaces that contain static information are referred to as read-only tablespaces, No media recovery is required except in some special cases, such as:

- When the last backup of the read-only tablespace was taken when the file was read-write and made read-only afterwards.
- When the last backup of the read-only tablespace was taken when it was read-only, was changed to read-write, and changed back to read-only.

These two situations are critical recovery processes requiring redo logs and the RECOVER command. All other situations are non-critical.

Password File

The standard method of authenticating to an Oracle 10g database by and database administrator is to log directly into the operating system of the server, connect directly to the database with Inter-Process Control (IPC). And establish a local connection on the database, not using SQL*Net. This process requires a password by the operating system account and allows the SYSOPER and SYSDBA accounts to be used.

Another method of authentication is by connect remotely using SQL*Net and a password file. This is the process used by the Enterprise Manager (EM). The ORAPWD utility defines the password used to connect to the database remotely as SYSOPER and SYSDBA.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

User Error Recovery

The Flashback Technologies used to recover from user errors include:

- Flashback Drop
- Flashback Versions Query
- Flashback Transaction Query
- Flashback Table

Overview of Technologies

All Flashback Technologies are designed to recover from user error or logical corruption:

- Flashback Table will recover a table when a table is updated with the incorrect WHERE clause.
- Flashback Versions Query will see the individual changes in the rows from query to recover an individual row or column within the row.
- Flashback Drop will undrop a table that has been accidentally dropped.
- Flashdrop Transaction Query can perform diagnostics on the changes which occur in a table or query.

Using Flashdrop Drop

The Flashback Drop technology provides a process of saving a copy of the dropped database object and dependent object in the Recycle Bin to allow recovery if required. The dropped database object is retained in the Recycle Bin until it is emptied and can be recovered without the need for an incomplete recovery, import, or use Log Miner. It is a faster method for recovery and will not impact other users of the database.

The Recycle Bin is a logical container fro storing all dropped tables and their dependent objects, such as indexes, constraints, triggers, nested tables, LOB segments, and LOB index segments.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

By default, the Recycle Bin is enabled in Oracle 10g. Database objects are received when the DROP TABLE command is executed. To recover these objects from the Recycle Bin, the Flashback Drop capability must be used. It requires more space utilization in the database since the dropped objects are still being stored after they have been dropped. The amount of space used to store dropped objects still counts against the tablespace quota.

After a table is dropped, it can be seen in the Recycle Bin along with its dependent object. These objects will be renamed using an unique naming convention to support the dropped objects of the same name by different users in case multiple users drop a table with the same name. The naming convention consists of a global UID, a unique, 24-character long identifier, and a version number assigned by the database. The Recycle Bin name for a dropped object is always 30 characters in length.

To view the Recycle Bin contents directly, use the special command, SHOW RECYCLE BIN. The dynamic views, USER_RECYCLEBIN and DBA_RECYCLEBIN, can be created by querying the Recycle Bin. From here, the use of Flashback Drop can proceed. Within one of these views, specify the OBJECT_NAME and use the command, FLASHBACK TABLE table_name TO BEFORE DROP. To rename the table from the original name, use the RENAME TO table_name option.

To permanently remove an object from the Recycle Bin and reallocate the space being consumed, use the PURGE TABLE original_table_name command. An object c an also be removed using the PURGE TABLE recycle_bin object_name command. All objects from a specific tablespace can be removed by using the PURGE TABLESAPCE command. Adding additional options for PURGE TABLESAPCE tablespace USER user, the tablespace contents for a specified username can be removed. The command DROP USER user CASCADE will drop a specific user and all associated objects. To purge the entire Recycle Bin, use:

- PURGE RECYCLE – to purge the commanding user's recycle bin.
- PURGE DBA_RECYCLE – to purge all users' recycle

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

bin.

Limitations of Flashback Drop and the Recycle Bin and impact their use:

- The functions or the Recycle Bin are only available for non-system, locally managed tablespaces.
- The timeframe that an object is stored in the Recycle Bin is determined by the system activity impacting space utilization, so there is no guarantee on how long a particular object is stored.
- Objects in the Recycle Bin cannot use Data Manipulation Language (DML) or Data Definition Language (DDL).
- To query a table in the Recycle Bin, its original name is will not be successful since the Recycle Bin name for the table is required.
- All dependent objects are retrieved when a Flashback Drop is performed, except when space pressure forces the removal of some dependent objects.
- The Virtual Private Database (VPD) and fine-grained auditing (FGA) policies on tables are not protected because of security concerns.
- Partitioned index-organized tables are not protected by the Recycle Bin.

Flashback Dropped Tables

Using the Enterprise Manager, the Flashback Dropped Tables capability can be performed. This is done by navigating through Maintenance > Recovery > Dropped Objects where the targeted database objects that have been Flashback Dropped can be identified.

Flashback Versions Query

The Flashback Versions Query allows all versions of the rows that exist or have existed between executed queries to be retrieved up to a determined point-in-time. All the committed occurrences, not uncommitted, of the rows for the query of an

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

object will be returned. This technology is an improvement to the Flashback Query originally designed for Oracle 9i.

The process of the Flashback Versions Query involves the technology retrieving data from the UNDO tablespace. The time in which undo data is stored in the database is determined in seconds by the UNDO_RETENTION initialization parameter. Normally, Oracle will start reusing undo space when transactions need additional undo space and none is available in the UNDO tablespace. The RETENTION GUARANTEE tablespace and database option can be set in the UNDO tablespace and protect unexpired undo data from being overwritten.

The privileges required to use the Flashback Versions Query are SELECT and FLASHBACK for each user. The function is performed with a new clause attached to the query statement, VERSIONS BETWEEN.

There are two implementation of VERSIONS BETWEEN: specify the system change numbers (SCNs) or a timestamp to identify a start and stop point. The clause can be used with the SCN MINVALUE and MAXVALUE options and the TIMESTAMP TO_TIMESTAMP option. THE SCN_TO_TIMESTAMP function can be used to identify the timestamp of an SCN.

Flashback Transaction Query

The Flashback Transaction Query is a diagnostic tool to identify changes made to the database at a transaction level. Analysis on data can be performed using this tool to be used for audits of transactions.

The technology is based on undo data. The time of the retained committed undo data is determined by the UNCO_RETENTION initialization parameter. The RETENTION GAURANTEE option is also used to protect the undo data from being overwritten.

To perform the analysis and diagnostics, perform a query of the view FLASHBACK_TRANSACTION_QUERY. The data from the view can assist in highlighting analysis of a specific transaction

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

or specific time.

Flashback Table

The Flashback Table allows a table or set table to be recovered from a specific point-in-time without performing an incomplete recovery. All dependent objects are also recovered. There are some significant benefits over an incomplete recovery which include:

- Faster and easier use.
- Availability of the database is not impacted.
- Users, not DBA, can quickly recover from logical corruptions.

Block Corruption

Block corruption occurs when a block on a physical disk becomes unreadable or inconsistent to the point where the data is unusable. Also called physical corruption, there are several sources of this form of corruption and can be very hard to diagnose. The use of software, firmware, and hardware bugs are some of the human error situation that can cause block corruption, as well as the environment housing the Oracle database.

Symptoms of Block Corruption

To detect block detection, multiple diagnostic logs are utilized to identify that corruption has occurred, including monitoring and reading of log files from the operating system, application, and database level.

System logs are different on each operating system. For Windows, the Event Viewer contains the contents of the system log and is accessible through Control Panel > Administrative Tools > Computer Management.. In Unix, the system logs events can be found in the SYSLOG file located usually in

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

/var/adm/syslog.

Application logs vary based on the application that is installed and the application administration guide should be consulted for the actual location. Though application logs will not provide detailed information about block corruption, they do provide significant information about any problems with application processes and procedures to identify the extent or impact of block corruption.

Database logs and trace files references are located in the ALERT>LOG file and the trace files in the UDUMP and BOUMP directories. The ALERT.LOG will identify problems at the highest level, while the trace files will provide detailed information about an ALERT.LOG entry to identify corruption problems.

Detecting and Resolve Corruption

Four methods exist for detecting corruption:

- The command, ANALYZE TABLE table_name VALIDATE STRUCTURE
- The Oracle DBVERIFY utility
- The init.ora parameter DB_BLOCK_CHECKING
- The DBWS_REPAIR package

The ANALYZE TABLE table_name VALIDATE STRUCTURE command can be used to validate the integrity of the object structure being analyzed. The result of the command will be successful or not successful at the object level. If an error is returned for the object analyzed, it will need to be completely rebuilt. If no error is returned, the object is free of corruption.

The DBVERIFY utility can be executed through the command prompt and can contain nine parameters:

- FILE – identifies the datafile to be verified.
- START – identifies the starting block to begin verification.
- END – identifies the ending block to begin verification.
- BLOCKSIZE – specifies the block size of the

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

database.

- LOGFILE – identifies the log file to store verification results.
- FEEDBACK – displays progress of the utility during verification process.
- PARAFILE – used to store options instead of using in command line.
- USERID – username and password.
- SEGMENT_ID – segment identifier.

For block checking at the database level, use the DB_BLOCK_CHECKING initialization parameter. For all non-system tablespaces, the default setting is false and the SYSTEM tablespace is enabled by default. This parameter is set dynamically with the ALTER SYSTEM SET statement and requires checks for corrupt blocks each time blocks are modified at the tablespace level. As a result a checksum occurs each time the block is modified.

The BMS_REPAIR package enables corrupt blocks to be detected and fixed in tables and indexes using a set of procedures performing several functions:

- CHECK_OBJECT – detects and reports corruptions in a table or index.
- FIX_CORRUPT_BLOCKS – marks blocks as software corrupt.
- DUMP_ORPHAN_KEYS – reports index entries pointing to rows in corrupt data blocks.
- REBUILD_FREELISTS – rebuilds freelists of the object.
- SEGMENT_FIX_STATUS – provides the capability to fix corrupted state of a bitmap entry.
- SKIP_CORRUPT_BLOCKS – ignores blocks marked corrupt during table and index scans.
- ADMIN_TABLES – provides administrative functions for repair of orphan key tables.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Block Media Recovery

Block Media Recovery (BMR) is a feature of RMAN to recover an individual block or group of corrupted blocks in a database. BMR is a supplement to datafile recovery, not a replacement and best used when minimal block corruption is present.

The advantages that BMR has over recovering the whole datafile include:

- A reduction in mean time to recovery (MTTR).
- Database operations are unaffected.
- Recovery is possible when redo logs are missing.

Some limitations of BMR include:

- Must be used with RMAN.
- The complete recovery of individual blocks must be performed.
- Corrupted blocks can never be recovered from the media backup source.
- A full RMAN backup must be available, not an incremental backup.
- Media corrupt blocks are not accessible to users until the recovery is complete.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Automatic Database Management

Within Oracle 10g is a new feature called Common Manageability Infrastructure (CMI) which improves the applications ability to be self-managed and self-tuning. The CMI is composed of:

- Automated Workload Repository (AWR)
- Server-generated alerts
- Automated routing maintenance features
- Advisory framework
- DBA components for automation and intelligent design

The AWR is used to collect and process performance statistics and metrics that can be used to detect problems and perform self-tuning. It is also the central storage repository for all elements of the CMI.

Server -generated alerts provide proactive information about database events and metric levels. Alerts are generated when metric threshold levels are exceeded. Some alerts will also send notifications through electronic mail or pager, execute operating system programs or PL/SQL scripts, or generate SNMP traps.

To handle time-consuming tasks automatically, there are several tasks related to routine maintenance. This is possible as the automatic statistics collection and automatic DML monitoring features collect and maintain optimizer statistics in real-time. The Automatic Database Diagnostic Monitor (ADDM) analyzes performance statistics in order to identify and resolve system bottlenecks. Automatic SQL tuning optimizes SQL tuning functions.

The advisory framework analyzes and provides recommendation in several key administrative areas. To simply usage and allows easy interactions, the advisors share a common interface.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Automatic Workload Repository (AWR)

At the core of the CMI is the Automatic Workload Repository which provides services to collect, process, maintain, and access database performance statistics and metrics for the database. The two elements that compose the AWR are:

- Statistics collection facility
- Workload repository

The statistics collection facility is responsible for collecting dynamic performance statistics which are stored in fixed tables accessible only from fixed views. These statistics are also stored in the workload repository.

Database performance statistics are used to measure the performance of the database over time to identify any problems and resolve. These statistics are dynamic, being initiated at startup and lost at shutdown. They are stored in dynamic performance tables, not the data dictionary like optimizer statistics. These dynamic performance tables are also called fixed tables which are memory structures that emulate tables to allow queries and views. The statistics fall into one of three categories:

- Cumulative values – statistics captured for the whole system, individual sessions or SQL statements, segments, and services which are cumulative over time.
- Metrics – represents the rate of change in cumulative statistics, measured in time, database calls, transactions, and other fixed units.
- Sampled data – a sampling of the current state of all active sessions.

Several new statistics are available in Oracle 10g and viewable by specific commands:

- V$SYS_TIME_MODEL command allows viewing of time model statistics:
    - CB CPU
    - BD TIME
    - JAVA EXECUTION ELAPSE TIME
    - PL/SQL COMPILATION ELAPSED TIME

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

    - PL/SQL EXECUTION ELAPSED TIME
    - BACKGROUND CPU TIME
    - BACKGROUND ELAPSED TIME
    - CONNECTION MANAGEMENT CALL ELAPSED TIME
    - FAILED PARSE ELAPSED TIME
    - FAILED PARSE (OUT OF SHARED MEMORY) ELAPSED TIME
    - HARD PARSE ELAPSED TIME
    - HARD PARSE (BIND MISMATCH) ELAPSED TIME
    - HARD PARSE (SHARING CRITERIA) ELAPSED TIME
    - INBOUND PL/SQL RPC ELAPSED TIME
    - PARSE ELAPSED TIME
    - SEQUENCE LOAD ELAPSED TIME
    - SQL EXECUTE ELAPSED TIME
- V$0SSTAT command allows viewing of operating system statistics:
    - AVG_BUSY_TICKS
    - AVG_IDLE_TICKS
    - AVG_IN_BYTES
    - AVG_OUT_BYTES
    - AVG_SYS_TICKS
    - AVG_USER_TICKS
    - BUSY_TICKS
    - IDLE_TICKS
    - IN_BYTES
    - NUM_CPU
    - OUT_BYTES
    - RSRC_MGR_CPU_WAIT_TIME
    - SYS_TICKS
    - USER_TICKS
- V$SERVICE_STATS command allows viewing of wait statistics:
    - APPLICATION WAIT TIME
    - DB CPU
    - CONCURRENCY WAIT TIME
    - USER COMMITS
    - USER I/O WAIT TIME
    - CLUSTER WAIT TIME

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

The workload repository adds persistence to the statistics collection facility. This is possible through the provision of a set of tables where performance data can be stored and accessed by other CMI components. In regular intervals, the Manageability Monitor (MMON) process transfers cumulative statistics in memory to the workload repository tables on disks to ensure survival in possible crashes or from being overwritten by newer statistics.

The workload repository ensures historical data is available for baseline comparisons, trend analysis, troubleshoot intermittently occurring problems. The data found in this repository is owned by the SYS user and stored in the SYSAUX tablespace. Each of the tables for the workload repository have the prefix, WR attached to their name. Examples of the tables found include:

- WRH$_ACTIVE_SESSION_HISTORY_BL
- WRH$_BG_EVENT_SUMMARY
- WRH$_BUFFER_POOL_STATISTICS
- WRH$_CLASS_CACHE_TRANSFER_BL
- WRH$_CR_BLOCK_SERVER
- WRH$_CURRENT_BLOCK _SERVER
- WRH$_DATAFILE
- WRH$_DB_CACHE_ADVICE_BL
- WRH$_DLM_MISC_BL
- WRH$_ENQUEUE_STAT
- WRI$_OPTSTAT_HISTGRM_HISTORY
- WRI$_OPTSTAT_HISTHEAD_HISTORY
- WRI$_OPSTAT_IND_HISTORY
- WRI$_OPSTAT_OPR
- WRI$_OPSTAT_TAB_HISTORY
- WRI$_SCH_CONTROL
- WRI$_SCH_VOTES
- WRI$_SQLSET_BINDS
- WRI$_SQLSET_DEFINITIONS
- WRM$_BASELINE

To enable the AWR, the STATISTICS_LEVEL initialization parameter must be set to TYPICAL or ALL. By using the option, BASIC, statistics must be manually gathered. An average

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

system using AWR will consume 200 to 300 MB of data for 10 concurrent active sessions assuming the default retention period of 7 days. This number is the product of the number of active sessions, the snapshot interval, and the retention period. Space consumption can be reduced by increasing the snapshot interval or decreasing the retention period.

Active Session History

The Active Session History (ASH) makes keeping statistics on the current session activity possible. It is a first-in, first-out (FIFO) buffer in memory collecting statistics on current session activity. These statistics are gathered by extracting samples from V$SESSION every second. Old statistics are continually dropped to make room for new ones.

The ASH resides in the System Global Area (SGA). Its size is fixed for the lifetime of an instance and calculated using the lesser of number of CPUs times 2MB of memory or 5 percent of the Shared Pool size.

The types of data sampled by ASH are:

- SQL_ID
- SID
- Client ID
- Service ID
- Program
- Module
- Action
- Object
- File
- Block
- Wait event number
- Actual wait time

These statistics in ASH can be viewed in the V$ACTIVE_SESSION_HISTORY fixed view. Each row in the view represents an active session. The data in this view represents a unique set of statistics and stores some of them in

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

a workload repository for persistent storage. There are two possible processes available to capture ASH data for the workload repository. The first process involves the MMON process flushing all data in the ASH buffer every 30 minutes. During this time, some of the key data is filtered into the AWR. If the ASH fills up in less than 30 minutes, the buffer is still flushed and key data filtered into the AWR; but this time, the effort is controlled by the Memory Monitor Light (MMNL) process.

Snapshots

The Enterprise Manager (EM) Database Control is the primary interface for the AWR through the Administration page. Under Workload, click the Workload Repository link to manage AWR settings and snapshots. The DBWS_WORKLOAD_REPOSITORY package is an API which allows management of all AWR functionality. The procedures available through this package include:

- CREATE_SNAPSHOT
- DROP_SNAPSHOT_RANGE
- CREATE_BASELINE
- DROP_BASELINE
- MODIFY_SNAPSHOT_SETTINGS

Performance Statistics are collected by the AWR by taking snapshots of the system at specific intervals. The snapshot pulls information from fixed tables holding performance statistics in memory. By default, snapshots are taken every hour and retain the snapshot for seven days before being automatically purged. The data from these snapshots are analyzed by the ADDM. To view the settings, use the DBA_HIST_WR_CONTROL view. To make changes to the settings, use the procedure, DBWS_WORKLOAD_REPOSITORY.MODIFY_SNAPSHOT_SETTINGS.

A unique snapshot ID is assigned to each snapshot which becomes a sequence number within the repository. This is true for all situations except when the Real Application Clusters (RACs) are used and every node within the cluster is queried.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Snapshots can be dropped using the DROP_SANPSHOT_RANGE procedure.

AWR Baselines

A baseline is a definition of a pair of snapshots which denote a significant workload period. The baselines are not aged out of the workload repository using the RETENTION settings, but are kept for historical comparisons to current system behavior. Baselines must be dropped manually. They can be used to define threshold settings for Oracle's server-generated alerts facility.

To create a baseline, use the CREATE_BASELINE procedure. It requires a beginning and ending snapshot ID, a name, and an optional database identifier (DBID).

AWR Views

A collection of data dictionary views can be used to obtain a comprehensive overview of both AWR information and configuration settings. The following views are available:

- DBA_HIST_ACTIVE_SESS_HISTORY
- DBA_HIST_BASELINE
- DBA_HIST_DATABASE_INSTANCE
- DBA_HIST_SQL_PLAN
- DBA_HIST_WR_CONTROL
- DBA_HIST_SNAPSHOT

AWR Reports

A standard summary report is available at any time against statistics stored in the AWR. One of two SQL scripts can be used to run the report:

- awrrpt.sql (text file)
- asrrpti.sql (HTML)

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

This report can be used to analyze system performance over a specified period of time. These scripts can be found in $ORACLE_HOME/rdbms/admin (Unix) or %ORACLE_HOME%\rdbms\admin (Windows).

Server Generated Alerts

Server-generated alerts are an element of the CMI that detects and notifies on problematic situations. The two classifications of alerts are threshold and non-threshold alerts. Threshold alerts utilize the computed metrics from the MMON process as a basis to determine the thresholds. If the threshold is exceeded, an alert is generated. Non-threshold alerts are triggered by database events. A set of default alerts are generated automatically by Oracle based on predefined criteria, or alerts can be defined by the DBA based on choices on metric thresholds.

Alert Process

The MMON process will automatically get computer metrics values from in-memory performance statistics every minute. Threshold values can be defined as wither warning, critical, or both. If this defined value is reached, the MMON process will compare the newly computed metric to the threshold level to determine if an alert must be generated.

With each generated alert, an alert message is sent to a persistent queue called ALERT_QUE. This queue is monitored by Enterprise Manager and alerts are displayed on the EM Database Control console. Notification options can be defined which can send alerts to designated administrators in one of several options. If an alert message cannot be written to the queue, it will be written to the alert log.

Configuring Thresholds

Thresholds can be managed through the EM Database Control

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Manage Metrics screen. To view current metric settings, open the Manage Metrics screen from the Database Control console. Each metric can be assigned a:

- Comparison operator
- Warning threshold
- Critical threshold
- Response action

Many metrics do not have defined thresholds by default. To set the levels for a metric, click the Edit Thresholds button on the Manage metrics screen. The settings available for defining threshold levels include the following options:

- Warning Threshold – a warning alert is generated when exceeded.
- Critical Threshold – a critical alert is generated when exceeded.
- Response Action – performs a response to a warning which can include:
    - Operating system command
    - PL/SQL procedures
    - SNMP trap

Multiple Thresholds can be set for distinct entities. To do so, select the metric and click on the Specify Multiple Thresholds button. The resulting screen is context-sensitive based on the selected metric. For instance if the metric refers to files, the screen will display all the individual files.

Alerts can be set up through the DBMS_SERVER_ALERTS PL/SQL package. There are two procedures in the package: SET_THRESHOLD and GET_THRESHOLD. The SET_THRESHOLD procedure will define the thresholds for a given metric, while GET_THRESHOLD will return the current threshold setting for a given metric. Thresholds are set with a comparison operator, with the following pre-defined constance being available by Oracle:

- OPERATOR_CONTAINS – value contained in the list of threshold values.
- OPERATOR_DO_NOT_CHECK – default threshold not applied to a specified object type.
- OPERATOR_EQ – metric value equal to threshold

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

value.

- OPERATOR_GE – metric value greater than or equal to threshold value.
- OPERATOR_GT – metric value greater than threshold value.
- OPERATOR_LE metric value less than or equal to threshold value.
- OPERATOR_LT – metric value less than threshold value.
- OPERATIOR_NE - – metric value not equal to threshold value.

The parameters that can be set using the SET_THRESHOLD include:

- METRICS_ID
- WARNING_OPERATOR
- WARNING_VALUE
- CRITICAL_OPERATOR
- CRITICAL_VALUE
- OBSERVATION_PERIOD
- CONSECUTIVE_OCCURRENCES
- INSTANCE_NAME
- OBJECT_TYPE
- OBJECT_NAME

Several views are available to access information on server alert, including:

- DBA_OUTSTANDING_ALERTS
- DBA_ALERT_HISTORY
- DBA_THRESHOLDS
- V$ALERT_TYPES
- V$METRIC
- V$METRIC_NAME
- V$METRIC_HISTORY

Automatic Routine Administration Tasks

Another element of the CMU is the automatic routine administration functionality which takes some of the most time-

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

consuming tasks and automates them.

Automatic Statistics Collection

Optimizer statistics can be generated manually using the DBMS_STATS packet or automatically using the automatic statistics collection functionality, which is recommended by Oracle. The different types of optimizer statistics include:

- Dictionary statistics
- System statistics
- Operating system statistics
- User-define statistics

The query optimizer for Oracle will choose the optimum plan for executing SQL statements against the database, but require accurate optimizer statistics for all objects that may be used to satisfy the SQL statement, including tables, columns, and indexes. The statistics differ based on the type of objects which are collected, such as:

- Table
  - Number of rows
  - Number of blocks
  - Average row length
- Columns
  - Number of distinct values in the columns
  - Number of nulls in the column
  - Histograms
- Indexes
  - Number of leaf blocks in index
  - Levels
  - Clustering factor
- System
  - I/O performance/utilization
  - CPU performance/utilization

When statistics on an object are collected, all preciously parsed SQL statements will be invalidate to force a new execution plan to be generated based on updated statistics.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Dictionary Statistics

Dictionary tables are heavily accessed and therefore a rich resource for fixed and real statistics, but having a great impact on overall system performance. The fixed tables reside only in memory with no associated I/O.

System Statistics

Optimal SQL execution plans require knowledge about CPU and I/O performance information which is available through system statistics. These statistics are classified as optimizer statistics, but previously parsed SQL statements are not invalidated when they are updated. Only new SQL statements will be parsed with new statistics. System statistics can be viewed using V$SYSSTAT or V$SESSTAT views.

Operating System Statistics

A special type of system statistics are the operating system statistics which provide data on CPU, memory, and file system utilization. These statistics are viewable through the V$OSSTAT. The different statistics that can be seen are:

- NUM_CPUS
- IDLE_TICKS
- BUSY_TICKS
- USER_TICKS
- SYS_TICKS
- IOWAIT_TICKS
- NICE_TICKS
- AVG_IDLE_TICKS
- AVG_BUSY_TICKS
- AVG_USER_TICKS
- AVG_SYS_TICKS
- AVG_IPWAIT_TICKS
- AVG_NICE_TICKS
- OS_CPU_WAIT_TIME
- RSBC_MGR_CPU_WAIT_TIME

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- IN_BYTES
- OUT_BYTES
- FS_IN_BYTES
- FS_OUT_BYTES
- AVG_IN_BYTES
- AVG_OUT_BYTES
- AVG_FS_IN_BYTES
- AVG_FS_OUT_BYTES

User Defined Statistics

Data within an Oracle is supported through an object model which allows users to define their own objects for data storage and manipulation and define new datatypes to supplement the native types found in the database.

Since most standard functions and index types will have difficulty with data stored in non-native format, the creation of functions are allowed by Oracle to support any new datatypes, and new user-defined statistics to support the optimizer.

Collecting Statistics

When a database is created in Oracle 10g, a job called GATHER_STATS_JOB is created in the scheduler whom runs whenever the Maintenance window is opened. If the Maintenance window closes before the job finishes, it will continue until it finishes. The default for opening the maintenance windows is weeknights from 10:00 PM until 6:00 AM and all day long on the weekends.

The DBMS_STATS.GATHER_DATABASE_STATS_JOB_PROC is an internal procedure used by the job to gather statistics. This procedure detects objects that have stale or no statistics. These objects are then prioritized and their statistics are collected. When called with the GATHER AUTO option, the procedure is similar to the DBMS_STATS.GATHGER_DATABASE_STATS procedure which doesn't perform any prioritization.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

To verify the job exists, query the DBA_SCHEDULER_JOBS view. The job can be disabled using the DBMS_SCHEDULE.DISABLE function.

Volatile Objects

Volatile objects can drastically change in size over the course of a day, such as tables supporting bulk-load operations, or tables that are truncated or drooped and rebuilt. These objects may not have any or inaccurate statistics. There are several options for dealing with volatile objects:

- Gather statistics manually using DBMS_STATS.
- Set statistics to NULL.
- Set and lock statistics to values typical for the table.

DML Table Changes

DML changes on tables are monitored using the initialization parameter STATISTICS_LEVEL as a global switch. It does not need to be enabled or disabled. DML activity, such as INSERT, DELETE, and UPDATE, is monitored against a table since the last time statistics were gathered. The System Monitor (SMON) process will update the data dictionary periodically with the gathered information.

Monitoring data can be viewed through:

- DBA_TAB_MODIFICATIONS
- ALL_TAB_MODIFICATIONS
- USER_TAB_MODIFICATIONS

The columns found in the ALL_TAB_MODIFICATIONS view include:

- TABLE_OWNER
- TABLE_NAME
- PARTITION_NAME
- SUBPARTITION_NAME
- INSERTS

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- UPDATES
- DELETES
- TIMESTAMP
- DROP_SEGMENTS

STATISTICS_LEVEL Parameter

The STATISTICS_LEVEL initialization parameter controls the automatic statistics collection functionality. The parameter can be set to one of three values: BASIC, TYPICAL, and ALL.

The BASIC setting disables the collection of several important statistics needed to support the functions of:

- ADDM
- Automatic optimizer statistics collection
- automatic SGA memory management
- AWR snapshots
- Buffer cache advisory
- Database time distribution statistics
- End-to-end application tracing
- Monitoring of statistics
- Object-level statistics
- PGA target advisory
- Segment-generated statistics
- Server-generation alerts
- Service-level statistics
- Shared Pool sizing advisory
- Timed statistics

The TYPICAL setting ensures all major statistics that is needed by the database for self-management are collected. IT is the recommended default for the parameter.

The ALL setting will gather all of the TYPICAL statistics and operating system and plan execution statistics.

To change the setting, use the ALTER SYSTEM setting at the system level or the ALTER SESSION statement at the session level. The only statistics affected at the session level are:

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Timed statistics
- Timed operating system statistics
- Plan execution statistics

The rest of the statistics are still gathered normally.

The statistics gathered at each level can be viewed using the V$STATISTICS_LEVEL view. The columns found in this view, include:

- DESCRIPTION
- SESSION_STATUS
- SYSTEM_STATUS
- ACTIVATION_LEVEL
- STATISTICS_VIEW_NAME
- SESSION_SETTABLE

Advisory Framework

The Advisory Framework is a consistent, uniform interface for advisors to interact with each other. The advisors are a collection of services that aid in the optimization of different aspects of the database. Each advisor covers a specific subsystem within the database will analyze, diagnose, and provide recommendations to eliminate performance issues in the database.

To support the framework, the AWR is utilized.

Advisors

The Automatic Database Diagnostic Monitor (ADDM) proactively identifies performance issues and bottlenecks within the database. It works with other advisors to create recommendations for resolving problems.

The Segment Advisor will analyze space fragmentation within segments and identify which segments can undergo an online shrink operation. It also reports on segment growth trends to assist in capacity planning and resource monitoring.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

The Undo Advisor aids in determining the size required for Undo tablespaces and optimal UNDO_RETENTION settings. The other functions of this advisor including assisting the prevention of Snapshot too old errors. Problems related to the Undo tablespace and advice related to their resolution is aided by the Undo Advisor.

The SQL Tuning Advisor will analyze individual statements and provide recommendations for increasing performance of the database. This advisor is run against:

- Problematic SQLs identified by the ADDM
- Resource intensive SQLs from the AWR
- Any user-defined SQL

The SQL Access Advisor analyzes a SQL workload and recommends access structures that can improve workload performance, including but not limited to, materialized views and indexes.

The Memory Advisor is used to tune the size of different memory structures used by Oracle. It works as a supervisory advisor responsible for the overall instance. Within the memory advisor are other advisors that focus on individual memory structures:

- SGA Advisor
- PGA Advisor
- Buffer Cache Advisor
- Library Cache Advisor

Invoking Advisors

Advisors can be invoked through several methods:

- EM Database Control
- MMON process
- Server Alert recommendations
- DBMS_ADVISOR package

Despite what method is used, there are common attributes can be shared:

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Mode of operation – advisors can run in LIMITED or COMPREHENSIVE mode.
- Time limit – identifies the maximum run time for the advisor.
- Interruptible – determines if advisors will return partial results in the event of an interruption.
- User directives – specifies if advisors will accept user directives.

DBMS_ADVISOR Package

The DBMS_ADVISOR package is an API that contains all the necessary constants and procedure declarations needed to access all the advisors. The procedures contained in this package are:

- CREATE_TASK
- DELETE_TASK
- EXECUTE_TASK
- INTERRUPT_TASK
- CREATE_TASK_REPORT
- RESUME_TASK
- UPDATE_TASK_ATTRIBUTES
- SET_TASK_PARAMETER
- MARK_RECOMMENDATION
- CREATE_TASK_SCRIPT

When tuning the operations of the database, the DBMS_ADVISOR utilizes these high level steps:

- CREATE_TASK – a data area set up in the repository to store and manage the task.
- SET_TASK_PARAMETER – allows parameters to be defined to control the advisor's behavior.
- EXECUTE_TASK – initiates synchronous execution of the task.
- CREATE_TASK_REPORT – creates a report of the analysis and recommendations from the advisor.
- MARK_RECOMMENDATION – allows recommendations to be accepted, rejected, or ignored.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- CREATE_TASK_SCRIPT – creates SQL script to implement accepted recommendations.

Advisor Views

To view information related to the advisors, the following views are available with the ADVISOR privilege:

- DBA_ADVISOR_DEFINITIONS
- DBA_ADVISOR_TASKS
- DBA_ADVISOR_LOG
- DBA_ADVISOR_PARAMETERS
- DBA_ADVISOR_COMMANDS
- DBA_ADVISOR_OBJECTS
- DBA_ADVISOR_FINDINGS
- DBA_ADVISOR_RECOMMENDATIONS
- DBA_ADVISOR_ACTIONS
- DBA_ADVISOR_RATIONALE
- DBA_ADVISOR_USAGE

Automatic Database Diagnostic Monitor

The ADDM is another key component of the CMI which provides automated proactive tuning of an instance in an Oracle 10g database. The ADDM analyzes all the statistics collected, identify a root cause for any found performance problems, and produce a report, and write an SQL to resolve any problems. This process is repeated every hour.

The benefits obtained by the ADDM include:

- An hourly production of the automatic performance diagnostic report.
- Problem diagnostics.
- Time-based quantification of impact and recommendation benefits related to problem.
- Rout causes of problems are identified.
- Recommendations for problem resolution.
- Identifies non-problems areas of the system.
- Minimal system overhead for diagnostics.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

The ADDM process is invoked automatically by the MMON process after each AWR snapshot is taken, which by default, is every hour. The analysis from the ADDM compares the most current snapshot to the prior one and identifies any existing bottlenecks, root causes, and non-problem areas of the system. The process relies on time model statistics which manages a common scale of comparison for all components. The DB_TIME statistics are key to the time model statistic and captures the total time spent in database calls for all components. It represents an aggregation of CPU and non-idle wait event time and is collected cumulatively from the startup time of an instance. The goal of tuning is to reduce DB_TIME.

Also beneficial to the ADDM is the Operation of the Wait Event model which enhances the granularity of the statistics. This granularity exposes a level of detail for individual areas of database operations. In addition, wait events are classified for easy identification. The different classifications are:

- Application
- Administration
- Commit
- Concurrency
- Network
- User I/O
- System I/O
- Configuration

After the ADDM analyzes the collected statistics, it returns a set of findings, recommendations, and rationales. The findings are a representation of the identified performance issues and classified as a symptom or root cause. Recommendations are suggestions for moving towards problem resolution. With each recommendation is the rationale behind it.

SQL Tuning Advisor

The SQL statement tuning is one of the biggest challenges in database administration. Oracle introduces automatic SQL tuning to resolve many of the issues found. The Automatic

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Tuning Optimizer (ATO) replaces the need for manually tuning and is designed in the query optimizer. What makes the ATO work successfully is the choice of tuning modes. The normal mode, a SQL statement is accepts and generates an execution plan based on the statistics available. The goal of this mode is to provide an execution plan quickly. In tuning mode, the primary goal is generating the best execution plan and therefore, performs a thorough analysis. It is recommended for complex, high-load statements that can have a great impact on system performance. The output of the ATO is not an execution plan, but a group of actions with their expected impact and rationale.

Automatic SQL Tuning will analyze SQL in four distinct methods:

- Statistics Analysis determines if essential statistics are missing or stale and make recommendations to gather relevant statistics. Any additional information is returned as a SQL Profile.
- SQL Profiling contains auxiliary statistics relevant to a single query to assist in generating an optimal execution path. This is done through sampling, partial execution, and execution history. They are applied to SELECT, INSERT, UPDATE, DELETE, CREATE TABLE AS SELECT, and MERGE statements.
- Access Path Analysis determines whether additional indexes could assist in increasing the performance of the query as well as a recommendation to run the SQL Access Advisor to determine the impact of the new index.
- SQL Structure Analysis reviews the syntax, semantics, and design of a query to determine common problems.

The SQL Tuning Advisor provides an interface to the ATO. The input into the SQL Tuning Advisor can come from several sources. A SQL Tuning Set (STS) allows groupings of SQL statements to be passed into the SQL Tuning Advisors,. A STS is a database object storing information on sets of SQL statements, such as:

- SQL text for multiple statements
- Execution context information

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

The scope and duration of a task performed by the SQL Tuning Advisor by one of the following options:

- Limited – performs analysis and returns recommendations, but not SQL Profile recommendations.
- Comprehensive – performs all functions in Limited mode, as well as, a full SQL Profile analysis.

The output of the SQL Tuning Advisor includes:

- Recommendations on optimizing the execution plan.
- Rationale for the recommendations.
- Estimated performance gains.
- Scripts for implementing the recommendations.

To access the SQL Tuning Advisor, the DBMS_SQLTUNE package is used. The package exposes the API to create and execute tuning tasks and process recommendations from the tuning tasks.

Automatic Storage Management

The Automatic Storage Management (ASM) system introduced with Oracle Database 10g is a cluster file system used with stand-alone instances or Real Application Clusters (RAC). The system provides a vertically integrated subsystem used to encapsulate a file system, volume manager, and fault-tolerate environment. It works with other Oracle features to automatically spread I/O loads across available hardware devices.

ASM Architecture

The ASM creates disk groups, a composition of disk devices and files residing on the disk devices which are managed as a logical unit, to alleviate the burden of administration. Datafiles and other database structures are divided into extents and those are divided among all the disks in the disk group in order to improve performance and reliability. Database objects are mirrored

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

rather than the entire disk volume.

A key feature of the ASM is automatic rebalancing which allows new disk devices to be added to the disk group and have files automatically redistributed to the new device.

Two background processes exist to allow databases instances to use ASM disks: OSMB and RBAL. The OSMB process performs communication between the database and ASM instance. The RBAL performs any opening or closing of disks within the disk group on behalf of the database instance. The RBAL exists, but performs differently, in ASM and database instances.

ASM Instance

ASM instances are accessed differently than the traditional methods of accessing databases. To manage disk groups, a dedicated instance is required by ASM. This instance has a memory footprint in the range of 60 to 100 MB. It is automatically configured when the database file storage option is specified to use ASM. The disk group can be configured using the Oracle Universal Installer (OUI). After a database is created, both the regular and ASM instances are started.

An ASM instance as several unique characteristics:

- No data dictionary.
- No control file.
- Connections through SYS and SYSTEM uses operating system authentication only.
- Disk group commands are only valid from an ASM instance.
- Con be in MOUNT or UNMOUNT state, but never OPEN.

Since there is no dictionary, access to an ASM instance requires connecting as SYSDBA or SYSOPER. SYSDBA users can perform all ASM operations. SYSOPER users have a limited set of commands available, including:

- Start up and shut down of an ASM instance.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Disk group mounting or dismounting.
- Altering status of disk group to ONLINE or OFFLINE.
- Disk group balancing.
- Performing integrity checks.
- Accessing V$ASM_* dynamic views.

ASM Initialization Parameters

Several initialization parameters exist for or in ASM instances. An SPFILE is recommended over an initialization parameter file for an ASM instance. There parameters include:

- INSTANCE_TYPE – has a value of ASM, though the default is RDBMS.
- DN_UNIQUE_NAME – has default value of +ASM and is a unique name for the group of ASM instances within a cluster or single node.
- ASM_POWER_LIMIT – controls the speed in which rebalance operations occur.
- ASM_DISKSTRING – limits the disk devices used to create disk groups.
- ASM_DISKGROUPS – specifies a list of disk groups to be automatically mounted by the ASM instance.
- LARGE_POOL_SIZE – defines the size of the pool.

Dynamic Performance Views

The more common ASM-related dynamic performance views include:

- V$ASM_DISK
- V$ASM_DISKGROUP
- V$ASM_FILE
- V$ASM_OPERATION
- V$ASM_TEMPLATE
- V$ASM_CLIENT
- V$ASM_ALIAS

Some of these views are also found within a RDBMS instance, though with different results.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

ASM Filenames

All files in ASM are OMF. When an object in an ASM disk group is dropped, the file is automatically deleted. The actual filename is not needed for most administrative functions and can be exposes through certain commands and data dictionary views. ASM filenames can take on one of six formats:

- Fully Qualified Name
- Numeric Name
- Alias Name
- Alias with Template Name
- Incomplete Name
- Incomplete Name with Template

Fully Qualified names are used when referencing an existing file and have the format:

+group/dbname/file type/tag.file.incarnation

Numeric names are used to reference an existing ASM file by disk group name and file number/incarnation pair. The format of the name follows:

+group.file_number.incarnation

Alias names reference existing objects or used to create a single ASM file. Aliases can be generated by the user with the ALTER DISKGROUP ADDALIAS command and do not end in a file number/incarnation pair. Alias with Template names are used only when creating new ASM files.

Incomplete names can be used for single file or multiple file creation operations. To generate the name, only the disk group name is used. A default template is used depending on the type of file. When generating an incomplete name with a template, the template name determines the characteristics of the file.

File Types and Templates

All files used by the database except for operating system executables are supported by ASM. These include:

- Control Files

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Datafiles
- Online Logs
- Archive Logs
- Temp Files
- RMAN Database Backup Piece
- RMAN Incremental Backup Piece
- RMAN Archive Log Backup Piece
- RMAN Datafile Copy
- Initialization Parameters
- Broker Config
- Flashback Logs
- Change Tracking Bitmap
- Auto Backup
- Data Pump Dumpset
- Cross-platform Datafiles

The default ASM file templates used by these file types are:

- CONTROLFILE
- DATAFILE
- ONLINELOG
- ARCHIVELOG
- TEMPFILE
- BACKUPSET
- XTRANSPORT
- PARAMETERFILE
- DATAGAURDCONFIG
- FLASHBACK
- CHANGETRACKING
- AUTOBACKUP
- DUMPSET

When creating a new disk group. A set of ASM file templates are copied from the default templates. The characteristics of these templates can be changed and applied to individual disk groups.

ASM Disk Groups

ASM Disk Groups improve I/O performance, increases

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

availability, and allows easier addition of disks to a disk group or an entire disk group. Disk groups are a collection of physical disks managed as a logical unit. Files in a disk groups are striped using either coarse or fine striping. Coarse striping will distribute files in units of 1MB across all the disks and is appropriate for a system with a high degree of concurrent small I/O requests. Fine striping distributes the files in units of 128kb and used in traditional data warehouse environments or OLTP systems.

When defining the type of mirroring to be used within a disk group, disks need to be grouped into failure groups, a set of one or more disks within a disk group that share a common resource. If this resource is to fail, the entire set of disks will be unavailable to the group. Once the failure group is defined, the mirroring of the disk group can be defined as one of these types:

- External redundancy – requires only one failure group and assumes that either the disk is not critical to the ongoing operation of the database or is managed externally with high-availability hardware.
- Normal redundancy – provides two-way mirroring and requires at least two failure groups in each disk group.
- High redundancy – provides three-way mirroring and requires at least three failure groups within a disk group.

Mirroring is managed at the extent level, not disks.

To view all disks discovered using the initialization parameter ASM_DISKSTRING and their status, use the view V$ASM_DISK.

Some of the operations that can be performed on disk groups are using the ALTER DISKGROUP statement:

- Adding a disk to a disk group.
- Dropping a disk from a disk group.
- Undropping a disk from a disk group.
- Rebalancing an ongoing disk group operation.
- Dismounting a disk group.
- Checking internal consistency of a disk group.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Recovery with ASM

Because ASM files cannot be accessed through the operating system, RMAN must be used to move database objects from a non-ASM disk location to an ASM disk group. The process involves:

1. Note filenames of control files and online redo log files.
2. Shut down the database using NORMAL, IMMEDIATE, or TRANSACTIONAL keywords.
3. Backup database.
4. Edit the SPFILE to use OMF for all file destinations.
5. Edit the SPFILE to remove the parameter, CONTROL_FILES.
6. Run the following RMAN with the correct filenames:

```
STARTUP NOMOUNT;
RESTORE CONTROLFILE FROM <controlfile location>;
ALTER DATABASE MOUNT;
BACKUP AS COPY DATABASE FORMAT
+<disk group destination>;
SWITCH DATABASE TO COPY;
SQL “ALTER DATABASE RENAME <logfile1>
TO '+<disk group destination>' “;
# repeat for all log file members
ALTER DATABASE OPEN RESETLOGS;
```

7. Delete or archive old database files.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Globalization Support

Support for globalization involves handling communication and transaction issues related to:

- Language
- Time Zones
- Currency
- Calendars

Features for Globalization

Globalization support from Oracle is a collection of features for managing data in multiple languages within the same database instance. These features include character sets and datatypes required to store multilingual. They convert dates, times, currency, numerics, and calendar data to follow locale conventions. Even utilities and error messages are translated into different languages. The underlying architecture is the National Language Support Runtime Library (NLSRTL).

Globalization support does not:

- Translate text into different languages.
- Does not control how multilingual text is displayed on client machines.

Given these limitations, the features that are available include:

- Language Support – stores, processed, and retrieved in most scripted languages.
- Territory Support – uses the NLS_TERRITORY parameter to define the cultural conventions which differ between geographical locations.
- Linguistic Sorting and Searching – provides accurate case conversion, sorting, and searching in all supported languages. Includes case-sensitive and accent-sensitive sorts and searches.
- Character Sets and Semantics – supports character sets based on national and international standards.
- Calendars – supports seven calendar systems: Gregorian, Japanese Imperial, Republic of China

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Official (ROC), Thai Buddha, Persian, English Hijrah, and Arabic Hijrah.

- Locale and Calendar Customization – allows customization of globalization definitions using the Locale Builder utility, including language, character set, territory, and linguistic sorting.

Architecture for Globalization Support

At the heart of Oracle's Globalization Support is the NLSRTL which offers a set of language-independent text and character-processing functions. When the database starts up, the NLSRTL attempts to find the file lx1boot.nlb, which defines the locale definitions available to the database. To find this file, an ORA_NLS10 variable is looked for in the environment to identify the right path.

Locale definition files reside in the same directory as the lx1boot.nlb file. The types of locale definition files and the maximum number available include:

- Language – 66 files
- Territory – 120 files
- Character set – 86 sets
- Linguistic sort – 8 sorts

Globalization Support is modular, ensuring that only the locale definition files required to support the database are used. Locale definitions can be mixed, matched, and modified without affecting other files.

Locale-dependent operations are governed by NLS parameters and NLS environment variables which are set on both the client and server sides. This allows multi-tier and client/server application support in any language. When a database is created, two session-independent NLS parameters are specified: a database character set and national character set.

When a connection is made to the database server by a client the character sets that exist on the client and server are compared. If no match is found, an automatic data conversion is

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

performed which is involves a lot of overhead. The only time this conversion is not performed is when the database character set is a strict superset of the client character set, which happens when:

- The superset contains all the characters defined in the subset.
- The encoded values of all characters in the subset match their encoded values in the superset.

Any application connecting to the server is considered a client and therefore governed by client-side NLS parameters. All local NLS operations are executed using environment variable settings initializing the client NLS environment. Local NLS operations are independently performed separate from any Oracle server session.

All known written languages in the world are represented in the universal character set, Unicode. This special character set provides a unique number for every character that can be used on any platform, any program, and any language.

NLS Parameters

NLS parameter settings define globalization support options. These settings can be specified in a number of ways and each method has a different effect. On the server, the NLS parameters are read from initialization parameter settings when the instance starts up. These values are stored in the data dictionary. On the client, the NLS parameters are defined as environment variables or set at the session level using the ALTER SESSION statement. Another method of defining these parameters is inside SQL function calls.

In most environments, setting the client side parameter, NLS_LANG, can meet the globalization requirements, since Oracle will automatically identify lower-level specifics from high-level settings. NLS parameters fall into the following classifications”

- Language and territory parameters
- Data and time parameters

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- Calendar parameters
- Numeric, list, and monetary parameters
- Length semantics

The NLS_LAND parameter defines the language, territory, and character set for the client. Setting this parameter is similar to setting the individual NLS_LANGAUGE, NLS_TERRITORY, and NLS_CHARACTERSET parameters. The following globalization elements can have their default conventions specified for the NLS_LANGUAGE parameter:

- Language for server messages
- Day and month names and abbreviations
- Symbols to represent AM, PM, AD, and BC
- Sorting sequence for character database
- Affirmation and negative response strings

The following globalization elements can have their default conventions specified for the NLS_LANGUAGE parameter:

- Date format
- Decimal characteristic
- Group separator
- Local currency symbol
- ISO currency symbol
- Dual currency symbol
- First day of the week
- Credit/debit symbols
- ISO week flag
- List separator

Data and time functions can be defined through the following individual parameters:

- NLS_DATE_FORMAT
- NLS_DATE_LANGUAGE
- NLS_TIMESTAMP_FORMAT
- NLS_TIMESTAMP_TZ_FORMAT

NLS Views

NLS setting information is stored in the data dictionary and inside

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

fixed tables residing in memory. Information about these settings cover the session, instance, and database. Several views are available to see the parameter values.

- NLS_SESSION_PARAMETERS
- NLS_INSTANCE_PARAMETERS
- NLS_DATABASE_PARAMETERS
- V$NLS_VALID_VALUES

Linguistic Sorts and Searches

Different rules are applied to each language when sorting text. Several results can be accomplished through text sorting. Some of the considerations for sorting are:

- Case-sensitive
- Accent marks (diacritics)
- Phonetics
- Character appearance
- Character groups

Oracle offers binary or linguistic sorts. Binary sorts are the fastest and efficient sorting method, but are limited. They perform numeric
sorts on the encoded value provided for each character. The character set simply needs to encode all the characters in the proper order.

The following list is characters and values for the US7ASCII character set:

A	65	a	97
B	66	b	98
C	67	c	99
D	68	d	100
E	69	e	101
F	70	f	102
G	71	g	103
H	72	h	104
I	73	i	105
J	74	j	106
K	75	k	107
L	76	l	108

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

M	77	m	109
N	78	n	110
O	79	o	111
P	80	p	112
Q	81	q	113
R	82	r	114
S	83	s	115
T	84	t	116
U	85	u	117
V	86	v	118
W	87	w	119
X	88	x	120
Y	89	y	121
Z	90	z	122

Linguistic sorts rely on the rules of specific languages to define sorting schemes rather than the values placed on the individual characters.

Though the linguistic sort definitions provided by Oracle covers most languages, the ability to define and modify new and existing definitions using the Oracle Locale Builder. The following elements are available and usable to create a comprehensive linguistic sort:

- Base letters – individual letters which are used to create other characters. A is a base letter and can be used to create À Á Â Ã a à á ã.
- Ignorable characters – diacritics and certain punctuation characters which have no effect on sort orders.
- Contracting characters – two or more characters which are treated linguistically as one character, for example the Spanish ch string.
- Expanding character – repeating or commonly occurring strings of characters that are compressed in locales into a single character.
- Context-sensitive characters – controls sorting based on order of characters, generally modified by the preceding character.
- Canonical equivalence – when using a Unicode character set, characters displayed differently but be

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

equivalent, for instance õ and o~.

- Reverse secondary sorting – strings containing diacritics are sorted left or right on base characters and right to left from right to left.
- Character rearrangement – sorting rules where certain characters switch position before sorting.

Linguistic Parameters

Linguistic sorts are applicable to specific languages or character set. The NLS_SORT parameter defines the type of sorting used to perform SQL sort operations. The default value for this parameter is defined by the language identified in the NLS_LANGUAGE parameter. To define linguistic sorting, the parameter is set to the name of any valid linguistic sort definition. Changes can be made to the default functionality, such as setting the default soft method for all ORDER BY operations and sort value for the NLSSORT function.

Not all SQL functionality or NLS-specific SQL functions support linguistic sorting. Some functions support only binary sorting capabilities. The methods that support linguistic sorting are:

- ORDER BY
- BETWEEN
- CASE WHEN
- HAVING
- IN/OUT
- START WITH
- WHERE

The NLS_COMP parameter operates with NLS_SORT to make linguistic sorting easier. When the parameter is set to a value of ANSI, the above SQL methods default to linguistic sorting, making the need to call the NLSSORT function unnecessary when using these operations. The available values for this parameter are BINARY and ANSI.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Linguistic Sort Types

The number of languages involved in the sort can determine the method used to perform the sort. A monolingual linguistic sort involves only one language in the sort. A two step process is used to compare character strings. The first step compares the major value of the strings, while the second step compares the minor string. Base letters and the characters derived share a common major value, but have different minor values.

Multilingual linguistic sorts involve two or more languages within the same sort. Three levels of evaluation are performed: primary, secondary, and tertiary. Primary sorts assign a primary value on the base letter of each character, ignoring case and diacritics. Ignorable characters are assigned a primary value of zero. Secondary sorts consider diacritics to differentiate accented letters by assigning a secondary sort level. Tertiary sorts consider case to differentiate between uppercase and lowercase letters.

By default, Oracle will consider both character case and diacritics when performing sort operations. This functionality can be overridden when required. For case-sensitive linguistic sorts, use the _CI string with the NLS_SORT parameter. The _AI string can be used for accent-sensitive linguistic sorts.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Database Resource Manager

Oracle offers the Database Resource Manager (DRM) as a component-based approach to manager resource allocations. There are three distinct elements of the DRM:

- Resource Consumer Groups
- Resource Plans
- Resource Pan Directives

Resources are allocated though resource consumer groups based on a resource plan. The plan consists of resource plan directives specifying how resources should be distributed among the groups. Resource consumer groups act as categories where user sessions can be assigned. Resource allocation methods are used to allocate resources such as CPU usage, sessions, idle time, operation execution time, and the like.

Pending Area

A pending area must first be established before defining or updating any DRM objects. It acts as a staging area where resource management objects can be defined and validated before being activated. To create a pending area, execute the procedure, DBMS_RESOURC_MANAGER.CREATE_PENDING_AREA. Once the pending area is created, all created are automatically stored until they can be validated and submitted.

To validate changes stored in the pending area to verify will not result in a violation in one of the following rules:

- A loop does not occur in the plan.
- All DRM objects identified actually exist.
- All plan directives refer to a plan of resource group.
- Allocation percentages for any given level do not exceed 100.
- Deletes are not attempted on plans currently used by

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

an active instance.

- The following parameters are only set by plan directives that refer to consumer resource groups:
  - ACTIVE_SESS_POOL_P1
  - MAX_EST_EXEC_TIME
  - MAX_IDLE_BLOCKER_TIME
  - MAX_IDLE_TIME
  - PARALLEL_DEGREE_LIMIT_P1
  - QUEUEING_P1
  - SWITCH_ESTIMATE
  - SWITCH_GROUP
  - SWITCH_TIME
  - SWITCH_TIME_IN_CALL
  - INDO_POOL
- No more than 32 resource consumer groups are contained in an active plan schema.
- Resource consumer group names and plan names do not conflict.
- A plan directive must exist for the OTHER_GROUPS group to allo9cate resources for sessions not identified in the active plan.

Validation of these changes can happen using the procedure, DBMS_RESOURCE_GROUP.VALIDATE_PENDING_AREA. To make changes active, use the DBMS_RESOURCE_MANAGER.SUBMIT_PENDING_AREA. Submitting the change involves three actions: validation, submission, and clearing the pending area. To clear the pending area without submitting any changes, use the DBMS_RESOURCE_MANAGER.CLEAR_PENDING_AREA procedure.

Resource Consumer Groups

Resource consumer groups allow users to be classified into logical groupings based on the requirements by the business or resource consumption. The DBMS_RESOURCE_MANAGER package allows resource4 consumer groups to be created, deleted, and modified to meet these requirements. To create a resource consumer group, use the procedure,

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

DBMS_RESOURCE_MANAGER.CREATE_ CONSUMER_GROUP. A unique name and description are required when created. The resource scheduling method that is used between sessions within a resource group can be defined using the CPU_MTH parameter with this procedure. Valid values for this parameter are ROUND_ROBIN and RUN_TO_COMPLETION.

Four predefined resource consumer groups are defined by default in a database:

- DEFAULT_CONSUMER_GROUP
- OTHER_GROUP
- SYS_GROUP
- LOW_GROUP

The DBMS_RESOURCE_MANAGER.UPDATE_CONSUMER_ GROUP procedure can be used to modify any consumer resource group. Within this procedure, the consumer group name, comments or description, and CPU resource allocation method can be changed.

Deleting a consumer resource group can be performed with the DBMS_RESOURCE_MANAGER.DELETE_CONSUMER_GRO UP procedure. When a group is deleted, the users assigned to that group will be assigned to the DEFAULT_SONSUMER_GROUP group. The current sessions assigned to the deleted group will be switched to the DEFAULT_CONSUMER_GROUP group.

User Session Assignments

Consumer groups can be automatically assigned to sessions based on specific session attributes in a process called consumer group mapping. To create a consumer group mapping, the procedure, DBMS_RESOURCE_MANAGER.SET_CONSUMER_GROUP_ MAPPING can be used to map sessions to consumer groups based on login or runtime session attributes. The valid session attributes that can be assigned are:

- CLIENT_OS_USER

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

- CLIENT_PROGRAM
- CLIENT_MACHINE
- MODULE_NAME
- MODULE_NAME_ACTION
- ORACLE_USER
- SERVICE_NAME
- SERVICE_MODULE
- SERVICE_MODULE_ACTION

Mappings define the session attribute, attribute value, and consumer group. Multiple consumer groups can be mapped to a session. Mapping priorities can be created using the procedure, DBMS_RESOURCE_MANAGER.SET_MAPPING_PRIORITY. These priorities are used to resolve any mapping conflicts between consumer groups.

Consumer groups can be explicitly changed for currently active user sessions using the SWITCH_CONSUMER_GROUP_FOR_USER procedure to change all active sessions for a particular user or SWITCH_CONSUMER_GROUP_FOR_SESS procedure to assign an active session to a new consumer group. Privileges can be granted to users to change their own consumer group. If this is done, the user can use the DBMS_SESSION.SWITHC_CURRENT _CONSUMER_GROUP procedure.

Resource Plans

Resources can be allocated across several resource consumer groups based on a resource plan, which consists of directives that specify how resources are distributed between resource consumer groups or other resource plans. The resource plan will allow resources to be allocated through the use of levels. Level 1 is the highest priority and level 8 is the lowest priority.

Simple resource plans allot only CPU resources to a small number of consumer groups. They are also known as single-level resource plans because no sub-plans exist. Complex resource plans can use any predefined resource allocation

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

methods. They include up to 32 consumer groups and multiple sun-plans.

There are tow options related to CPU allocation methods: EMPHASIS and RATIO. The EMPHASIS method is the most common method used and used for either single- or multi-level plans. In this method, percentages are used to represent CPU resource allocations. The RATIO method is only available for the single-level plans and CPU resource allocations are represented as a weight in plan directives.

The DBMS_RESOURCE_MANAGER.CREATE_SIMPLE_PLAN procedure allows a simple resource plan to be created. Up to eight consumer groups and their CPU allocations can be created. The DBMS_RESOURCE_MANAGER.CREATE_PLAN procedure allows complex resource plans to be created. Each created plan requires a plan name to be defined, a comment or description, and the resource allocation methods. Several parameters used to create complex resource plans include:

- PLAN
- COMMENT
- CPU_MTH
- ACTIVE_SESS_POOL_MTH
- PARALLEL_DEGREE_LIMIT_MTH
- QUEUEING_MTH

Resource sub-plans and resource plans are created the same way and have no differences. They are sub-plans simply because a higher-level pan allocates resources to it.

Modifying resources plans is possible through the procedure, DBMS_RESOURCE_MANAGER.UPDATE_PLAN. To verify the values for the parameters of the resource plan, any of the following views can be used:

- V$ACTIVE_SESS_POOL_MTH
- V$PARALLEL_DEGREE_LIMIT_MTH
- V$QUEUEING_MTH
- V$RSRC_PLAN_CPU_MTH

Resource plans can be deleted using the DBMS_RESOURCE_ MANAGER.DELETE_PLAN procedure which will leave all

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

subordinate objects intact. The DBMS_RESOURCE_ MANAGER.DELETE_PLAN_CASCADE procedure will delete a subordinate plan.

Resource Plan Directives

To create complex resource plans, resource plan directives are required. They assign consumer groups to resource plans and define the resource allocation. They work by specifying the owning resource plan, the target consumer group or sub-plan, and the assigned resource allocations, which are managed by setting parameters to ensure the right target has the proper resources. The resource allocation methods are predefined by Oracle and cannot be modified.

Below are the available resource allocation methods:

- CPU – specifies how CPU resources are allocated between consumer groups or sub-plans.
- Active session pool with queuing – limits the number of concurrent active sessions are available to a consumer group.
- Degree of parallelism limit – specifies the maximum parallel degree for any operation within a consumer group.
- Automatic consumer group switching – allows sessions which have exceeded certain execution time criteria are dynamically switched to a different group.
- Canceling SQL and termination sessions – specifies any long-running queries and sessions will be terminated automatically if the execution time threshold is exceeded.
- Execution time limit – specifies the maximum estimated execution time which will be allowed for any operation.
- Undo pool – specifies the amount of undo that can be generated by a consumer group.
- Idle time limit – specifies the maximum amount of time a session can remain idle.
- To create a resource plan directive, use the procedure, DBMS_RESOURCE_

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

MANAGER.CREATE_PLAN_DIRECTIVE. The additional parameters available to amend the procedure are:

- PLAN – name of the resource plan which the directive belongs.
- GROUP_OR_SUBPLAN – name of the consumer group or sub-plan which is being allocated resource by this directive.
- COMMENT – description of plan directive.
- CPU_Px – represents either CPU allocated at level 1 or the ratio weight for CPU resources. When using the EMPHASIS method, x represents additional levels for CPU percentages up to 8.
- ACTIVE_SESS_POOL_P1 – specifies the maximum number of concurrently active sessions. The default, NULL, represents unlimited sessions.
- QUEUEING_P1 – number of seconds before a job in the inactive session queue times out. The default, NULL, means the job never times out.
- PARALLEL_DEGREE_LIMIT_P1 – maximum degree or parallelism defined for any operation. NULL represents no imposed limit.
- SWITCH_GROUP – identifies the consumer group which this sessions will be switched if the criteria is met.
- SWITCH_TIME – identifies the number of seconds that a session can execute an operation before a group switch will occur.
- SWITCH_ESTIMATE – requests an estimated execution time for an operation before the execution begins which is compared to the SWITCH_TIME parameter. If the estimate exceeds the parameter setting, the switch is performed before execution.
- MAX_EST_EXEC_TIME – requests an estimated execution time for an operation before the execution begins which is compared to the SWITCH_TIME parameter. If the estimate exceeds the parameter setting, an ORA-07455 error occurs and the operation does not perform.
- UNDO_POOL – defines the maximum KB of undo that can be generated by the consumer group/sub-

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

plan.

- MAX_IDLE_TIME – defines the number of seconds that a session can remain idle before the session is killed.
- MAX_IDLE_BLOCKER_TIME – defines the number of seconds that a blocking session can remain idle before a session is killed.
- SWITCH_TIME_IN_CALL – defines the number of seconds that a session can execute before a group switch occurs.

To update resource plan directives, use the procedure, DBMS_RESOURCE_MANAGER.UPDATE_PLAN_DIRECTIVE. It uses the same parameters for creating a plan directive. To delete a directive, use the DBMS_RESOURCE_MANAGER.DELETE_PLAN_ DIRECTIVE procedure.

Automating Tasks

Scheduler

To schedule routine tasks to execute at specific dates and times, Oracle 10g introduced a new Scheduler function. With this feature, it is possible to manage recurring tasks that run at preset intervals or one-time tasks set to execute immediately or in some distant point in time.

The Scheduler utilizes several components:

- Jobs – instructions to the Schedule to run a specific program at a specific date and time.
- Programs – the code that must be executed to accomplish a task and containing parameters that are passed to the program at runtime. Programs can be stored as an object which is accessible by several jobs.
- Schedules – contains a start date, optional end date, and a repeat interval. Schedules can be stored as

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

independent object that can be used by several jobs.

- Windows – a recurring block of time where a specific resource plan is to be enabled to govern resource allocation for the database.
- Job groups – a logical method for classifying jobs with similar characteristics. Jobs within the job group will inherent specific classifications after being associated.
- Window groups – a logical method of managing windows by allowing members windows to be manipulated as one object.

Scheduler Architecture

The architecture of the Schedule consists of three elements:

- Job Table – stores all the active jobs within a database.
- Job Coordinator – a process to ensure that jobs run on schedule.
- Job Slaves – processes that ensure the execution of the job under the guidance of the job coordinator.

The job table stores information about all jobs, including the objects referenced by the job, the job owner, and the next run date, as well as statistical information about jobs including how many times the jobs have been run and/or failed. The *_SCHEDULER_JOBS view allows information stored in the table to be viewed.

The job coordinator will queue the job table regularly and copies the information stored there into a memory cache for improved performance when creating the job. This information is passed to the job slave for execution. The database will monitor job schedules and initiated the job scheduler process when required if the process has not already started.

Every aspect of the job slave pool within the database is controlled by the job coordinator, allowing dormant slave processes to be removed and spawning new processes when needed. A job slave will receive job information from the job

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

coordinator and proceed to collect all the metadata needed to carry out the request, such as program arguments and privilege information. A new session is then created and a transaction started within the session. The job is executed and when complete, the transaction is committed and the job slave closed. The slave performs the actions:

- Update the STATUS column to a COMPLETED value in the job table.
- Update the RUN_COUNT column to an increment of 1 in the job table.
- Insert an entry into the job table.
- Search for new work to be performed

The architecture differs slightly between a stand-alone database environment and a RAC environment. The difference in an RAC environment includes:

- Each instance in the cluster has its own job coordinator.
- Al the job coordinator communicates between themselves to share information.
- Definitions of job can used service affinity rather than instance affinity, allowing situations where a service consists of multiple instances can still run the job when one of the instances is down.

Administration Tools

The DBMS_SCHEDULER is a PL/SQL package that offers procedures used to create and manage scheduler objects. The procedures include:

- ENABLE
- DISABLE
- SET_ATTRIBUTE
- SET_ATTRIBUTE_NULL

All Scheduler objects, except schedules, have the common attribute ENABLED. The attribute is BOOLEAN in style and identifies if an object can be used by the Scheduler. All objects, except schedules are disabled by default. Schedule objects

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

cannot be disabled and therefore enabled automatically when created. To enable any of the other objects, use the DBMS_SCHEDULE.ENABLE procedure. Only one argument, NAME, is acceptable to the procedure. The argument allows either the name of a single object, of a list of objects separated by commas.

When enabling group objects:

- When a job group is enabled, all members of the job group are enabled.
- When a window group is enabled, only the window group object is enabled, not the individual members of the windows group.
- When a window or window group is referenced in the ENABLE procedure, it must be prefixed with the SYS schema name.

The DBMS_SCHEDULER.DISABLE procedure can be used to disable a Scheduler object, which sets the ENABLED attribute to FALSE. The parameters accepted by the procedure are NAME and FORCE. The name parameter is the same for enabling the object. The FORCE parameter is another Boolean style parameter that instructs the procedure on how to handle dependencies which are classified as:

- A job object that references a program object.
- An instance of an object that is currently running.

If no dependencies are found, any valid Scheduler object will be disabled. Any dependencies found will determine the outcome of the DISABLE procedure based on the object. Some special cases related to disabling object include:

- Windows that are disabled will not disable the jobs referencing the group.
- Window groups that are disabled do not affect members of the group.

Setting Attributes

Scheduler objects are collections of attributes. To change an object requires setting attributes, using the

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

DBMS_SCHEDULER.SET_ATTRIBUTE or DBMS_SCHEDULER.ATTRIBUTE_NULL procedure. The first procedure will set an attribute for any type of Scheduler object, while the second procedure sets any attribute to NULL and useful to unset the attribute. When setting an attribute, three parameters are accepted:

- NAME
- ATTRIBUTE
- VALUE (not used for the NULL statement)

Scheduler Jobs

A job in the Scheduler defines a specific program to be executed, the arguments used by the program, and the schedule defining when to execute the program. Jobs are created using the DBMS_SCHEDULER.CREATE_JOB procedure. Arguments that may be required by the job are based on the program being passed on to the job and can be set using the SET_JOB_ARGUMENT and/or SET_JOB_ANYDATA_VALUE procedures. Job attributes which control certain behaviors of the job may be required and can be set using there the CREATE_JOB procedure or inherited through the job classes which the jobs are assigned to. The attributes available to jobs include:

- JOB_NAME – name assigned to the job.
- JOB_TYPE – the type of job: either PLSQL_BLOCK, STORED_PROCEDURE, or EXECUTABLE.
- JOB_ACTON – allows execution of program external to the database.

Job Classes

A job class is a container object for logically grouping jobs into a larger unit. Using job classes allow easier management of small numbers of job groups. They can be assigned to a resource consumer group and allow jobs to be prioritized within them. They can have a specific set of attributes to be used to define the characteristics of the class. The attribute parameters for job

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

classes include:

- JOB_CLASS_NAME
- RESOURCE_CONSUMER_GROUP
- SERVICE
- LOGGING LEVEL
- LOG_HISTORY
- COMMENTS

The CREATE_JOB_CLEASS procedure can be used to create job classes. Dropping a job class is possible in the DROP_JOB_CLASS procedure.

Calendaring

Jobs to be executed must be scheduled. A specific calendaring syntax allows flexibility in defining repeating intervals for jobs. The elements of the syntax include:

- FREQ – defines the frequency type as yearly, monthly, weekly, daily, hourly, by the minute or second.
- INTERVAL – defines how often the recurrence repeats. If the FREQ is set for daily, a INTERVAL value of 1 will runt he job each day, while a value of 2 will run the job every other day.
- BYMONTH – specifies the month(s) to execute the job.
- BYWEEKN0 – specifies the week of the year as a number to execute the job.
- BYYEARDAY -specifies the day of the year as a number to execute the job.
- BYMONTHDAY – specifies the day of the month as a number.
- BYDAY – specifies the day of the week as a three-letter abbreviation.
- BYHOUR – specifies the hour of the day to execute the job.
- BYMINUTE – specifies the minute of the hour to execute the job.
- BYSECOND -specifies the second of the minute to

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

execute the job.

When creating schedules, the following rules should be followed:

- Frequency must always be the first element defined.
- All other elements must follow the frequency element and can be in any order.
- Elements should be separated by semicolons and represented only once.
- List of values within an element should be separated by commas and do not need to be in order.
- Calendaring statements are case-sensitive.
- Whitespace is allowed.
- The BYWEEKNO element can only be used when the FREQ element is set to YEARLY.
- The BY DAY element is used when the FREQ element is set to YEARLY or MONTHLY.
- To set the last day of the month, use BYMONTHLY=-1.
- Monday is always considered the first day of the week.
- Time zones or Daylight Savings Time adjustments are not allowed.

Scheduler Windows

Windows can be created using the DBMS_SCHEDULER.CREATE _WINDOW procedure. When creating a window, either an existing schedule can be used or an inline schedule used. The parameters available for the procedure include:

- WINDOW_NAME
- RESOURCE_PLAN
- START_DATE
- DURATION
- SCHEDULE_NAME
- REPEAT_INTERVAL
- END_DATE
- WINDOW_PRIORITY
- COMMENTS

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Windows can be opened using its schedule or manually with the OPEN_WINDOW procedure.

Window logs are maintained for all window activities. These logs can be viewed by the DMA_SCHEDULER_WINDOW_LOG procedure. The different activities that can be viewed include:

- Creating a new window
- Dropping a window
- Opening a window
- Closing a window
- Overlapping a window
- Disabling a window
- Enabling a window

Scheduler Views

The following views are available for displaying different elements of the Scheduler:

- *_SCHEDULER_SCHEDULES
- *_SCHEDULER_PROGRAMS
- *_SCHEDULER_PROGRAM_ARGUMENTS
- *_SCHEDULER_JOBS
- *_SCHEDULER_GLOBAL_ATTRIBUTES
- *_SCHEDULER_JOB_ARGUMENTS
- *_SCHEDULER_JOB_CLASSES
- *_SCHEDULER_WINDOWS
- *_SCHEDULER_JOB_RUN_DETAILS
- *_SCHEDULER_WINDOW_GROUPS
- *_SCHEDULER_WINGROUP_MEMBERS
- *_SCHEDULER_RUNNING_JOBS

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Questions

The following multiple-choice questions are a refresher.

Question 1

What is the command to automatically allow channel allocation during backup, restore, or delete?

A) CONFIGURE CHANNEL DEVICE TYPE DISK
B) CONFIGURE CHANNEL DEVICE TYPE TO DISK
C) CONFIGURE DEVICE CHANNEL TYPE DISK
D) CONFIGURE DEVICE TO DISK CHANNEL

Question 2

Which backup type utilizes a proprietary RMAN format?

A) Backup group
B) Image copy
C) Backup set
D) Image set

Question 3

RMAN dynamic views are stored in what object?

A) Recovery catalog database
B) Target database
C) Control files
D) Management database

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 4

What must be completed before running the ORAPWD utility to create a password file?

A) The database is started
B) The database is shut down.
C) The database is started in MOUNT mode
D) None of the above

Question 5

What is the process for recovering a control file?

A) Start up database.
Restore control file.
Start up database.
Recover the database.
Open database.
B) Start up database.
Restore control file.
Start up database in MOUNT mode.
Recover the database.
Open database.
C) Start up database in MOUNT mode.
Restore control file.
Start up database.
Recover the database.
Open database.
D) Start up database in NOMOUNT mode.
Restore control file.
Start up database in MOUNT mode.
Recover the database.
Open database.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 6

When performing a recovery in the Enterprise Manager, which of the following credentials is needed?

A) Any network account
B) Oracle account in Unix
C) Database account with SYSDBA privileges
D) Administrative account in Windows

Question 7

The Flashback Database is best suited for what type of recovery?

A) User error
B) Incomplete recovery
C) Physical corruption
D) Media failure

Question 8

How can it be verified that the Flashback Database is turned on?

A) Check the initialization parameter for the Flashback Database
B) Check the DATABASE_ON column in the DBA_FLASHBACK_DATABASE view
C) Check the FLASHBACK_ON column in the V$DATABASE dynamic view
D) Check the FLASHBACK_ON column in the DBA_FLASHBACK_DATABASE view

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 9

What can be performed using the Flashback Drop process?

A) Back up tables only
B) Back up tablespaces only
C) Back up indexes and tables only
D) Back up table and dependent objects

Question 10

What keyword must be specified when creating a disk group to allow a disk that has been previously used as part of another disk group to be reused?

A) USE
B) REUSE
C) FORCE
D) USE_DISK

Question 11

How many calendars are supported by Oracle 10g?

A) 1
B) 2
C) 7
D) 14

Question 12

What is the determining factor in resolving block corruption?

A) The tables and indexes impacted by the corruption
B) The extent of the corruption in the table(s)
C) The block(s) in the corrupted table
D) The tablespaces impacted by the corruption

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 13

What database technology is used by all the Flashback Technologies, except the Flashback Database to recover data?

A) Archive logs
B) Redo logs
C) Rollback segments
D) Undo data

Question 14

How is the DB_FLASHBACK_RETENTION_TARGET parameter measured?

A) Using the SCN
B) Using redo log sequence and threads
C) Using time in days
D) Using time in minutes

Question 15

Which of the following capabilities of an incomplete recovery is available to both RMAN and user-managed backup methods?

A) SET UNTIL TIME
B) UNTIL SEQUENCE
C) UNTIL TIME
D) SET TIME

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 16

RMAN-based database recovery goes by what other name?

A) Traditional recovery
B) Server-managed recovery
C) User-managed recovery
D) Database recovery

Question 17

Which of the following tablespace types contain static information?

A) Set tablespace
B) Index tablespace
C) Read-only tablebase
D) Temporary tablespace

Question 18

Before an incremental backup can be made, that type of backup must be performed first?

A) Level 0
B) Level 1
C) Level 2
D) Level 3

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 19

What is the term for the RMAN information stored in the control files which can be written over when necessary?

A) Reuse records
B) Circular reuse records
C) Writable information
D) Non-circular reuse records

Question 20

The Recovery Manager does not support which of the following backup capabilities?

A) Control file backups
B) Open database backups
C) Closed database backups
D) Password file backups

Question 21

Which method used when creating backup sets to improve backup performance?

A) Multiplex
B) Parallelism
C) Fragmentation
D) Multicopy

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 22

What can be used to monitor RMAN backups?

A) LIST commands
B) RMAN dynamic views
C) REPORT commands
D) All of the above

Question 23

Authentication and security of the user is accomplished in local connections to an Oracle database using which method below?

A) Database password
B) Password file
C) Operating system password
D) All of the above

Question 24

Which of the following is the best description of an incomplete recovery?

A) An incomplete recovery stops before the failure.
B) An incomplete recovery is missing transactions.
C) An incomplete recovery is missing selected files,
D) An incomplete recovery is missing data.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 25

What is the name of the background process responsible fro writing before block images and recovering from the Flashback Database log?

A) RVRW
B) RWVR
C) RVWR
D) RMRV

Question 26

What is the default setting for the size of the redo log file created in the flash recovery area?

A) 50MB
B) 100MB
C) 150MB
D) 200MB

Question 27

Which command should be used to remove the objects in multiple users from the Recycle Bin?

A) PURGE RECYCLEBIN
B) PURGE TABLESPACE_USER
C) PURGE TABLE user
D) PURGE DBA_RECYCLEBIN

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 28

What is the primary cause of block corruption?

A) Memory errors written to disk
B) Random I/O errors written to disk
C) Bugs introduced by hardware, software, or firmware
D) Corrupted data that updates several tables

Question 29

How many CPU allocation levels can be defined by a simple resource plan?

A) 2
B) 4
C) 8
D) 16

Question 30

What is the unique identifier for a SQL statement in the database?

A) SID
B) SSID
C) SQLID
D) SQL_ID

Question 31

What background process is used to coordinate rebalance activities for disk groups?

A) OSMB
B) RBAL
C) OSMB
D) None of the above

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 32

Which of the following is a valid ADMM finding type?

A) Information
B) Symptom
C) Problem
D) All of the above

Question 33

What is the Flashback Versions Query function?

A) A method for querying all versions changes on rows existing between the times the query is executed and a point-in-time in the past.
B) A method for querying all versions changes existing between the times the query is executed and a predetermined point-in-time in the past.
C) A query used to perform diagnostics on table changes in row data between the times the query was executed and a determined point-in-time.
D) A query used to perform administrative management of version changes to tables between times the query was executed and a point-in-time in the past.

Question 34

Which administrative database activity is not benefited by performing a Flashback Database recovery?

A) Dropping a table
B) Dropping a tablespace
C) Resizing datafiles to a larger size
D) Resizing datafiles to smaller size

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 35

Which of the following scenarios can be recovered using a Flashback Database?

A) A loss of a control file
B) An incomplete batch transaction
C) A user error resizing datafiles to a smaller size
D) A table dropped through RESETLOGS

Question 36

Which of the files listed below must be in the matching location of the ASCII control file in order to rebuild the control file?

A) Control files
B) PFILE or SPFILE
C) Redo logs
D) Datafiles

Question 37

Which of the following situations is considered a critical loss to the database?

A) Losing the current or active redo member
B) Losing the redo log group before being archived
C) Losing the current or active redo log group
D) Losing the archive file

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 38

What can be done to rebuild the objects in the index tablespace more efficiently?

A) Rebuild the index in parallel with NOLOGGING
B) Rebuild the index with LOGGING
C) Rebuild the index with NOLOGGING
D) Rebuild the index the parallel with LOGGING

Question 39

What is the ratio size can backups be compressed to?

A) 4-1
B) 5-1
C) 15 percent
D) 20 percent

Question 40

Which of the following commands can be used to display the settings of automatic channel setup?

A) SHOW CHANNEL
B) SHOW SETTING
C) SHOW ALL
D) SHOW CONFIGURATION

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Answer Guide

Question 1
Answer: A
Reasoning: When using the CONFIGURE command, the information is stored as a default and doesn't require to be specified when performing backups, restores, or deletes. With this in mind, the appropriate command is CONFIGURE CHANNEL DEVICE TYPE DISK.

Question 2
Answer: C
Reasoning: The Backup set utilizes the RMAN proprietary format. Within this format, only used blocks are backed up.

Question 3
Answer: B
Reasoning: RMAN dynamic views are stored in the target database.

Question 4
Answer: B
Reasoning: Before the ORAPWD utility can be used, the database must be shutdown

Question 5
Answer: D
Reasoning: When recovering a control file, the database must be started in NOMOUNT mode. The control file is restored and the database started in MOUNT mode. The control file can than be recovered and the database opened.

Question 6
Answer: B, C, D
Reasoning: Two credentials are required when running a recovery with Enterprise Manager: the operating system account and the database account. The operating system can be either an Oracle account in Unix or administrator account in Windows. The database account must have SYSDBA privilege.

Question 7

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Answer: A
Reasoning: The best situations for using a Flashback Database are user errors and logical corruption.

Question 8
Answer: C
Reasoning: The V$DATABASE dynamic view can be used to determine if the Flashback Database is on or off through the FLASHBACK_ON column.

Question 9
Answer: D
Reasoning: The Flashback Drop process is responsible for backing up tables and their dependent objects.

Question 10
Answer: C
Reasoning: FORCE is the keyword that must be specified to reuse a disk that was previously used by another disk group.

Question 11
Answer: C
Reasoning: Oracle 10g supports the Gregorian, Japanese Imperial, ROC Official, Persian, Thai Buddha, Arabic Hijrah, and English Hijrah calendars: 7 in all.

Question 12
Answer: B
Reasoning: The determining factor when resolving block corruption is the extent of the corruption, namely whether it exists in a few blocks or is more widespread.

Question 13
Answer: D
Reasoning: Undo data is used by the Flashback Drop, Flashback Table, and Flashback Versions Query to recover data, but not the Flashback Database.

Question 14
Answer: D
Reasoning: The parameter for

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

DB_FLASHBACK_RETENTION_TARGET is measured in minutes, determining how many minutes that data is written before the data is overwritten.

Question 15
Answer: C
Reasoning: The UNTIL TIME clause is available to both user-managed and RMAN backups.

Question 16
Answer: B
Reasoning: Server-managed recovery is another name for RMAN-based recovery.

Question 17
Answer: C
Reasoning: Read-only tablespaces contain static data that cannot be changed.

Question 18
Answer: A
Reasoning: Level 0 is a full backup that must be performed on all used blocks to create a baseline.

Question 19
Answer: B
Reasoning: Circular reuse records contain information that can be written over when the need arises.

Question 20
Answer: D
Reasoning: Backups of password files are not supported by the RMAN utility.

Question 21
Answer: A
Reasoning: Improved performance is possible using multiplexing by copying multiple database files at the same time and can be used with backup sets and image copies.

Question 22

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Answer: D
Reasoning: RMAN activities can be monitored differently there the LIST and REPORT commands and dynamic views.

Question 23
Answer: C
Reasoning: The operating system password is used to authenticate a user attempting to access the database from a local connection.

Question 24
Answer: A, B, D
Reasoning: An incomplete recovery is a recovery that stops before the failure and missing transaction. If is missing some data that was previously stored in the database before the failure.

Question 25
Answer: C
Reasoning: The RVWR process is responsible for writing the “before” image information to the Flash Database log which is used to perform a Flashback Database recovery.

Question 26
Answer: B
Reasoning: The default size for a redo log file created in the flash recovery area is 100MB.

Question 27
Answer: D
Reasoning: To purge the Recycle Bin for all users, use the PURGE DBA_RECYCLEBIN command.

Question 28
Answer: C
Reasoning: The majority of block corruption is rooted in the human introduction of bugs with new hardware, software, and firmware changes.

Question 29
Answer: C

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Reasoning: Up to eight levels of CPU allocation can be defined by a simple resource plan.

Question 30
Answer: D
Reasoning: SQL_ID is a hash value that uniquely identifies a SQL statement.

Question 31
Answer: B
Reasoning: The RBAL process coordinates rebalance activity within an ASM instance. The ORBn process is used to perform the extent movement of an ASM instance, while the OSMB acts as a bridge between the ASM and RDBMS instance.

Question 32
Answer: D
Reasoning: Analysis results from the ADDM fall in the categories of Problem, Symptom, and Information. A Problem addresses the root problem identified. Symptom refers to the symptom resulting from the root problem. Information addresses the non-problem areas.

Question 33
Answer: A
Reasoning: The best answer is that the Flashback Versions Query is a method for querying all version changes. The changes that are focused on are within the time limits of when the query is executed and a determined point-in-time in the past.

Question 34
Answer: D
Reasoning: Resizing datafiles to a smaller size cannot be undone using a Flashback Database.

Question 35
Answer: B
Reasoning: A Flashback Database recovery is useful for incomplete batch transactions or user errors such as truncated tables.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Question 36
Answer: D
Reasoning: Datafiles must be in the matching location. The server files, PFILE and SPFILE, must be available. The control files and redo logs can be rebuilt.

Question 37
Answer: B, C
Reasoning: The loss of the redo log group or the current/active redo group is considered critical losses for a database. Non-critical situations represent events where limited impact on database operations exists and an incomplete recovery is possible. These situations require more than is available through an incomplete recovery.

Question 38
Answer: A
Reasoning: The most efficient method is the rebuild the index in parallel using NOLOGGING. It has minimal impact on the operations of the database, but runs the risk of using server resources extensively.

Question 39
Answer: B, D
Reasoning: The ratio for compressed backups is 5-1 or 20 percent of the original file size.

Question 40
Answer: C
Reasoning: To view the automatic channel allocation settings, use the command SHOW ALL which will display all the configuration settings.

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

References

Dawes, Chip, Bryla, Bob, Johnson Joseph C., and Weishan, Matthew. *OCA: Oracle 10g Administration I Study Guide*, Sybex Inc., Alameda, California: 2005.
Stuns, Doug, Buterbaugh, Tim, Bryla, Bob. *OCA: Oracle 10g Administration II Study Guide*, Sybex Inc., Alameda, California: 2005.

OCA information: www.comptia.com

Websites

www.artofservice.com.au
www.theartofservice.org
www.theartofservice.com

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Glossary

A

ADDM (Automatic Database Diagnostic Monitor) 5, 104, 109, 118-20, 122-3, 172
alerts 89, 104, 111-13
Application Programming Interface (API) 70, 109, 121, 125
applications 11, 17-18, 28, 30, 100-1, 123, 134
architecture 2, 11, 87-8, 132, 148-9
archive logs 33, 64-5, 68, 73, 78, 91, 94, 129, 159
attributes 13, 49, 51-2, 55, 57, 59, 63, 76, 121, 149-51, 154
AWR (Automatic Workload Repository) 5, 22, 104-5, 107-10, 119-20

B

background processes 18, 20-1, 29, 35, 40, 88, 126, 163-4
backup sets 71, 74-80, 84, 155, 161, 168, 170
baseline 107, 109-10, 170
block corruption 5, 100-1, 164, 171
blocks 16, 35, 55, 73, 76-8, 80, 100-3, 107-8, 114, 151, 158, 168-70

C

catalog 8, 64-5, 68, 70, 74-5
channels 71-2, 76, 79
character sets 23, 31, 34, 36, 53, 132-6, 138
CMI (Common Manageability Infrastructure) 104-5, 111, 122
columns 13-15, 56, 58-61, 96, 114, 117, 119, 157, 169
components 33-6, 40, 65, 68, 123, 147
configuration 9, 25, 28, 71-2, 78, 85, 92, 123, 167
connect 20-1, 35, 37, 39, 41, 65, 68-70, 72, 74, 95
constraints 2-3, 14-15, 45, 52, 56, 58-61, 96
control files 3, 22-3, 31, 33, 38, 40, 64-7, 72-3, 76-7, 83-5, 88-90, 93, 126, 155-6, 166, 168
copy 24, 52, 66, 74-5, 84, 96, 131
CPU 106, 115-16, 123, 142, 144-6

D

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

data blocks 18, 45-6, 48, 55, 74, 78-9
data dictionary 13-14, 31, 45-7, 93, 105, 110, 117, 126, 128, 134, 136
database 3, 11-53, 55-60, 62-6, 68-70, 72-8, 82-97, 99-100, 102-3, 105, 116-22, 129-31, 133, 148, 156-7, 168-9
Database Administrator 1-2, 17, 95
Database Control 16, 28-9, 109, 112
database files 25, 32-3, 66, 74, 77, 83-4, 93, 131
database objects 35, 52-3, 58, 68, 97-8, 126, 131
database operations 93, 103, 123, 173
database recovery 4, 33, 83, 86, 160
datafiles 21, 23, 31-3, 38, 45-6, 49-52, 64-8, 74-8, 80-1, 83-4, 89, 101, 103, 107, 126, 129
datatypes 3, 53, 56-7, 59, 116, 132
DBA 14, 32, 35-6, 41, 47, 50, 83, 87, 89, 97, 100, 109-11, 113, 117, 122, 157
DBMS 5, 112, 114, 116-17, 121, 125, 140-5, 147, 149-51, 153
DEST 19-20, 44, 47, 89-91
dictionary 14, 46-7, 51, 127
directories 26, 33, 47, 53, 74, 77, 133
disk groups 32, 125-7, 129-31, 158, 164, 169
disks 21-2, 32-3, 37, 46, 66, 75-7, 83-4, 88, 107, 126-7, 130-1, 155, 158, 164, 169
display 26, 59, 72, 92, 112, 167, 173

E

EM (Enterprise Management) 16, 72, 83, 86, 92-3, 95, 109
EM Database Control 43, 51, 112, 120
environment 10, 20, 36, 61, 125, 133-4
execute 27, 76, 104, 121, 140, 146-7, 151-3

F

FILE 19-20, 47, 67, 81-2, 89-90, 101, 127
files 19, 22-6, 32-3, 37-9, 47, 50, 73, 77-8, 80-3, 88-91, 93, 95, 112, 126, 128-31, 133
flash recovery area 4, 33, 84, 88-92, 163, 171
Flashback Database 4, 87-8, 91-3, 157, 159, 163, 165-6, 169, 171-2

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

Flashback Versions Query 5, 87, 96, 98-9, 169, 172
functions 16, 56, 87, 93, 98-9, 102, 116, 118, 120, 125

I

image 4, 77-9, 88, 170
incremental backups 4, 76, 78-80, 103, 160
indexes 3, 5, 15, 45-6, 48, 56, 60-2, 93-6, 102, 114, 120, 124, 158, 167, 173
information 7, 14, 22-4, 28, 30-1, 38, 44, 50-1, 65, 67-8, 73-4, 80-2, 88-9, 101, 148-9, 172
initialization parameters 2, 19, 31, 34-6, 38-9, 44, 90, 102, 127, 129, 157
interface 21, 28, 66, 70, 124

J

jobs 20-1, 116-17, 146-54

L

languages 20, 36, 53, 132-9
level 58, 78-9, 112, 114, 117, 119, 123, 139-40, 143, 146, 160, 170
linguistic sorts 133, 136-9
LIST command 75, 81-2, 162
locations 10, 20, 22-3, 27, 31-4, 36-8, 44, 51, 68, 76, 81, 84, 88-90, 92, 101, 131
logs 8, 44, 51, 57, 72, 86, 88, 92, 95, 154

M

management 11, 21, 48-9, 109, 151
maximum number 20-1, 35, 46, 133, 146
memory 20, 25, 34-5, 63, 106-9, 115, 136
memory structures 11, 18, 40, 105, 120
metadata 2, 13, 51, 67, 149
metrics 104-5, 112-13
mode 36, 40-1, 124
monitor 17-18, 28-9, 31, 51, 92

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

N

NLS 6, 20, 132, 134-6, 138-9
NULL 58-9, 117, 146, 149, 151

O

objects 17, 36, 58, 63, 76, 97-9, 101-2, 108, 113-17, 122, 128, 147-50, 155, 163, 167
OMF (Oracle Managed Files) 19-20, 47, 51, 90, 128, 131
operating system 16, 24-6, 32, 35-6, 69-70, 95, 100, 118, 131, 168
operations 26, 46-8, 51, 57, 71-2, 93, 121, 123, 127, 130, 138, 145-6, 173
Oracle 11-14, 18-19, 25-7, 32-5, 38, 40, 45-6, 48, 53-6, 83-4, 87, 99, 104-5, 110-12, 114, 116
Oracle database 9, 11, 14, 24, 30, 33-4, 64-6, 87, 94, 100, 162
OSMB 126, 164
OUI (Oracle Universal Installer) 2, 26-7, 126
output 51, 81-2, 124-5

P

parallel 141, 144, 146, 167, 173
parameters 19, 33, 35, 39, 44, 71-2, 90-2, 101-2, 118, 121-2, 134-6, 138, 141-2, 144, 146-7, 150-1
plan 140-1, 144, 146-7
plan directives 140-1, 144
POOL 107, 127, 141, 144, 146
problems 40, 50, 101, 104-5, 107, 120, 122-3, 165, 172
process 17, 20-1, 32, 35, 40, 42-3, 62, 65, 73, 78-9, 81, 86, 94-6, 109, 122-3, 148
program 108, 134, 143, 147, 151, 154

Q

query 12, 15-16, 22, 51, 57, 62, 96, 98-9, 105, 117, 124, 165, 172
queue 20, 22, 111, 148

R

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

RACs (Real Application Clusters) 20, 109-10, 125
rationale 122-5
Reasoning 168-73
rebuild 62, 102, 166-7, 173
recommendations 26, 48, 104, 119-25
recover 23, 69, 74, 76, 81, 86, 88, 96-7, 100, 103, 156
recovery 4, 6, 11, 20, 22-4, 33, 40, 42-3, 52, 64-6, 81, 83-7, 89-93, 96, 103, 157
Recycle Bin 96-8, 163, 171
redo logs 24, 38, 49, 57, 66, 76, 90, 94-5, 103, 159, 166, 173
Relationships 2, 14-15
report 76-7, 81, 110-11, 121-2
resource allocation 145, 148
resource plans 6, 140, 143-6, 148, 164, 172
restoration 64-5, 83, 88
restores 65, 69, 71, 73, 76-7, 81, 83, 155, 168
RMAN (Recovery Manager) 3-4, 33, 50, 64-76, 79-81, 83-6, 89, 91-2, 103, 131, 155, 159, 161-2, 168
RMAN backups 64-6, 81, 85, 103, 170
RMAN utility 67-70, 72, 83, 170
rows 12-13, 15, 48, 55-8, 60, 81, 96, 98, 102, 108, 114, 165

S

schedules 147-9, 151, 153-4
schema objects 3, 45-6, 52-3, 56-7, 63
SCN (system change number) 75, 77, 85-6, 88, 92-3, 99, 159
scripts 40, 43, 65, 76, 111, 125
seconds 55, 99, 146-7
segments 2, 15-16, 21, 23, 45, 47, 49, 52, 62, 102, 105, 118, 120
sequences 3, 45, 56, 63, 76, 80, 85, 159
server 11, 21, 26, 30, 35, 68, 72, 86, 95, 104, 107, 112-13, 134
services 7, 37, 105, 119, 149
sessions 17, 21, 57-8, 76, 81, 105, 136, 140-3, 145-7, 149
shutdown 3, 39, 41-4, 76, 86, 91, 105, 168
snapshots 5, 109-10, 120, 123
sorting 21, 132, 136, 138
space 16, 18-19, 48, 62, 78-9, 89
SQL (Structured Query Language) 2-3, 12, 16, 42, 44, 57, 59,

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

76, 81, 92, 95, 108, 110, 122, 124, 131
start 3, 21, 27, 29-30, 37-40, 43, 63, 86, 94, 99, 101, 127, 138, 153, 156
statement 18, 46, 50, 57-9, 62-3, 120, 125
statistics 5, 22, 89, 105, 107-10, 114-19, 122-4
storage 15, 33, 35-6, 45, 51, 88
stores 23, 54-6, 61, 80, 109, 132, 148
support 31, 34, 64, 66, 92, 97, 116, 118-19, 132-3, 138, 161
syntax 42, 44, 71-2, 124, 152
system statistics 5, 114-15

T

tables 3, 12-15, 23, 45-7, 55-62, 65, 68, 96-8, 100, 102, 107, 114, 117-18, 148, 158, 164-6
tablespaces 3, 5, 21, 23-4, 28, 31, 45-53, 55, 57, 62, 64-5, 68, 75, 84, 93-5, 158
tape 66, 75, 83-4, 88, 91
target database 4, 65, 67-72, 75-6, 81, 83, 85, 155, 168
tasks 17, 39, 64, 104, 121-2, 125, 147
templates 6, 30-2, 36-7, 127-9
temporary tablespaces 4, 46-9, 51, 93-4, 160
time 39, 42-3, 51, 54, 71, 73, 75, 77-8, 91, 95, 99-100, 102-3, 105, 109-11, 145-8, 165
TIMESTAMP 54, 60, 99, 118, 135
tool 17, 23, 29-30, 39, 44, 64, 99
transactions 13, 24, 43, 49, 57, 85, 99, 105, 149
types 13-16, 19, 21, 27, 30-2, 36-7, 44, 59-60, 65, 71, 75, 78, 113-14, 127-8, 130, 151

U

Unix 25-6, 30, 100, 111
updates 23, 117-18, 149, 164
USER 14, 59, 68, 97, 106, 115, 117, 142-3, 163
user errors 23, 87, 96, 157, 169, 172
users 12, 14, 17-18, 22-4, 26, 40-3, 52-3, 68, 87, 96-7, 99-100, 103, 116, 128, 141-3, 171

V

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055

values 1, 8, 14-16, 39, 58, 60, 63, 91, 112-13, 117-18, 127, 134, 136-9, 144, 151-3
Virtual Private Database (VPD) 98

W

window 30, 150, 153-4
Windows 30, 37, 100, 111, 148, 150, 153-4, 168
workload repository 105, 107, 109-10

Copyright The Art of Service | Brisbane, Australia | Email:service@theartofservice.com
Web: http://theartofservice.com | eLearning: http://theartofservice.org | Phone: +61 (0)7 3252 2055