Oracle Database 19c: Backup and Recovery

**Table of Contents**

**Course Practice Environment: Security Credentials 5**

Course Practice Environment: Security Credentials

**Practices for Lesson 1: Backup and Recovery Overview 7**

Practices for Lesson 1: Overview

**Practices for Lesson 2: Backup and Recovery Configuration 9**

Practices for Lesson 2: Overview

Practice 2-1: Verifying that the Control File is Multiplexed

Practice 2-2: Configuring the Size of the Fast Recovery Area

Practice 2-3: Verifying that the Redo Log File Is Multiplexed

Practice 2-4: Configuring ARCHIVELOG Mode

**Practices for Lesson 3: Using Recovery Manager (RMAN) 27**

Practices for Lesson 3

Practice 3-1: Configuring the Default Backup Destination

Practice 3-2: Setting the Date and Time Format for RMAN

Practice 3-3: Configuring RMAN Settings

**Practices for Lesson 4: Backup Strategies 35**

Practices for Lesson 4: Overview

**Practices for Lesson 5: Creating Database Backups 37**

Practices for Lesson 5: Overview

Practice 5-1: Backing Up the Control File

Practice 5-2: Verifying Automatic Backups of the Control File and SPFILE

Practice 5-3: Creating a Whole Database Backup

Practice 5-4: Creating Partial Database Backups

Practice 5-5: Configuring Block Change Tracking

Practice 5-6: Using Incremental Backups

Practice 5-7: Backing Up Additional Database Files

**Practices for Lesson 6: Using Optional Backup Features 73**

Practices for Lesson 6: Overview

Practice 6-1: Creating an Archival Backup

**Practices for Lesson 7: Tuning RMAN Backup Performance 81**

Practices for Lesson 7: Overview

Practice 7-1: Monitoring an RMAN Backup Job

**Practices for Lesson 8: Recovery Catalog Overview 87**

Practices for Lesson 8

Practices for Lesson 9: Creating a Recovery Catalog 89

[Practices for Lesson 9: Overview](#_TOC_250029)

[Practice 9-1: Creating a Recovery Catalog Owner](#_TOC_250028)

[Practice 9-2: Creating the Recovery Catalog](#_TOC_250027)

Practices for Lesson 10: Managing Target Database Records 95

[Practices for Lesson 10: Overview](#_TOC_250026)

[Practice 10-1: Registering a Database in the Recovery Catalog](#_TOC_250025)

[Practice 10-2: Configuring the Recovery Catalog for Recovery](#_TOC_250024)

Practices for Lesson 11: Using Stored Scripts 105

[Practices for Lesson 11: Overview](#_TOC_250023)

[Practice 11-1: Creating a Stored Script](#_TOC_250022)

[Practice 11-2: Executing a Stored Script](#_TOC_250021)

[Practices for Lesson 12: Creating and Using Virtual Private Catalogs 113](#_TOC_250020)

[Practices for Lesson 12: Overview](#_TOC_250019)

[Practice 12-1: Enabling the Virtual Private Database (VPD) Functionality](#_TOC_250018)

Practices for Lesson 13: Restore and Recovery Concepts 125

[Practices for Lesson 13](#_TOC_250015)

Practices for Lesson 14: Diagnosing Failures 127

[Practices for Lesson 14: Overview](#_TOC_250014)

[Practice 14-1: Diagnosing and Repairing Database Failure](#_TOC_250013)

Practices for Lesson 15: Performing Complete Recovery 135

[Practices for Lesson 15: Overview](#_TOC_250012)

[Practice 15-1: Recovering from the Loss of a System-Critical Data File](#_TOC_250011)

[Practice 15-2: Recovering from the Loss of an Application Data File](#_TOC_250010)

Practices for Lesson 16: Performing Point-in-Time Recovery 165

[Practices for Lesson 16: Overview](#_TOC_250009)

[Practice 16-1: Recovering a Table from a Backup](#_TOC_250007)

Practices for Lesson 17: Performing Block Media Recovery 199

[Practices for Lesson 17: Overview](#_TOC_250006)

[Practice 17-1: Repairing Block Corruption](#_TOC_250005)

[Practices for Lesson 18: Performing Additional Recovery Operations 205](#_TOC_250004)

[Practices for Lesson 18: Overview](#_TOC_250003)

[Practice 18-2: Restoring the Control File](#_TOC_250001)

[Practice 18-3: Recovering from the Loss of All Control Files](#_TOC_250000)

Practice 18-4: Restoring the Password File

**Practices for Lesson 19: Oracle Flashback Technology Overview 231**

Practices for Lesson 19: Overview

**Practices for Lesson 20: Using Logical Flashback Features 233**

Practices for Lesson 20: Overview

Practice 20-1: Preparing to Use Flashback Technologies

Practice 20-2: Restoring a Dropped Table

Practice 20-3: Using Flashback Table

**Practices for Lesson 21: Using Flashback Database 245**

Practices for Lesson 21: Overview

Practice 21-1: Enabling Flashback Logging

Practice 21-2: Performing Flashback Database

**Course Practice Environment:**

**Security Credentials**

**Course Practice Environment: Security Credentials**

For operating system (OS) user names and passwords:

* Ask your instructor for credential information.

For product-specific credentials used in this course, see the following table:

|  |  |  |
| --- | --- | --- |
| **Product-Specific Credentials** | | |
| **Product/Application** | **Username** | **Password** |
| Database | SYS | fenago |
| Database | SYSTEM | fenago |
| Database | RCATOWNER | fenago |
| Database (ORCLCDB) | SYS | fenago |
| Database (ORCLCDB) | SYSTEM | fenago |

**Practices for Lesson 1:**

**Backup and Recovery Overview**

**Practices for Lesson 1**

There are no practices for Lesson 1.

**Practices for Lesson 2: Backup and Recovery**

**Configuration**

**Practices for Lesson 2: Overview**

### Overview

In these practices, you learn how to configure your database to enable recovery from various losses. You verify the control file configuration, the fast recovery area (FRA), redo log groups, ARCHIVELOG mode, and redundant archive log destinations.

How to configure your database for recovery:

* Ensure redundancy of control files. If a control file is damaged or lost, recovery is easier if you have another copy.
* Review the fast recovery area configuration.
* Ensure that there are at least two redo log members in each group. If a redo log member is damaged or lost, recovery is easier when you have an additional member in the group.
* Place your database in ARCHIVELOG mode. In all cases, you will be able to recover the database either completely or incompletely depending on which database files have been damaged or lost.
* Configure redundant archive log destinations. In cases where you lost archive log files and you need them to recover the database, you will be able to perform an incomplete recovery, unless you have a duplicate version of the archive log in another destination.

### Prerequisites:

* Ensure you have access to an Oracle 19c environment.
* You must have the necessary privileges to start the listener and the database.

**Steps:**

1. Open the Terminal: On your system, open the terminal (or command prompt) application.

2. Enter the following command as root user: **xhost +**

3. Switch to the Oracle User: To switch to the `oracle` user, enter the following command:

**su - oracle**

Password is not required for the oracle user.

4. Start the Listener: The listener is a crucial component in Oracle as it listens for incoming client connection requests and manages traffic to the database.

To start the listener, enter the following command:

**lsnrctl start**

**lsnrctl status**

Wait for a message indicating that the listener has successfully started.

**Practice 2-1: Verifying that the Control File is Multiplexed**

### Overview

In this practice, you verify that the control file is multiplexed.

A control file is a small binary file that describes the structure of the database. It must be available for writing by the Oracle server whenever the database is mounted or opened. Without this file, the database cannot be mounted, and recovery or re-creation of the control file is required. Your database should have a minimum of two control files on different storage devices to minimize the impact of a loss of one control file. The loss of a single control file causes the instance to fail because all control files must be available at all times. However, recovery can be a simple matter of copying one of the other control files. The loss of all control files is slightly more difficult to recover from, but is not usually catastrophic.

### Assumptions

You are logged in as the oracle user.

#### Time Estimate

It should take approximately 25 minutes to complete this practice.

### Tasks

1. Open a terminal window and use the oraenv script to set the environment variables for the

orclcdb container database.

$ **. oraenv**

ORACLE\_SID = [orclcdb] ?

The Oracle base remains unchanged with value /u01/app/oracle

$

1. Start SQL\*Plus and connect to the CDB root as the SYS user with the SYSDBA privilege.

$ **sqlplus / AS SYSDBA**

…

Connected to an idle instance. SQL>

1. If the database instance is not started, start the instance and open the database.

SQL> **startup**

ORACLE instance started.

…

Database mounted. Database opened.

SQL>

1. Find out how many control files exist in the database. The query returns the names of two control files (control01.ctl and control02.ctl), which verifies that the control files are multiplexed.

SQL> **SELECT name FROM v$controlfile;**

NAME

/u01/app/oracle/oradata/ORCLCDB/control01.ctl

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl SQL>

When the CDB was created, DBCA created two control files. When you use the CREATE DATABASE command in SQL\*Plus to create a database, you configure the CONTROL\_FILES parameter to generate two control files and set their names.

1. View the CONTROL\_FILES parameter. Notice that the paths to the control files are stored in this parameter. The results below are formatted for easier viewing.

SQL> **SHOW PARAMETER control\_files**

NAME

TYPE

VALUE

control\_files string

/u01/app/oracle/oradata/ORCLCDB/control01.ctl,

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl

SQL>

1. Create a parameter file (PFILE) from the server parameter file (SPFILE).

SQL> **CREATE PFILE FROM SPFILE;**

File created.

SQL>

1. Shut down the database instance in IMMEDIATE mode.

SQL> **SHUTDOWN IMMEDIATE**

Database closed. Database dismounted.

ORACLE instance shut down.

SQL>

1. Exit SQL\*Plus.

SQL> **EXIT**

1. Create a directory for the new control file.

$ **mkdir -p /u01/app/oracle/controlfiles\_dir/ORCLCDB**

$

1. Before you edit your PFILE, make a backup of it.

$ **cp $ORACLE\_HOME/dbs/initorclcdb.ora**

**$ORACLE\_HOME/dbs/backup\_initorclcdb.ora**

$

1. Copy one of the control files to the directory you created in a previous step (/u01/app/oracle/controlfiles\_dir/ORCLCDB) and name the file control03.ctl.

$ **cp /u01/app/oracle/oradata/ORCLCDB/control01.ctl**

**/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl**

$

1. Open the PFILE (initorclcdb.ora) in the editor of your choice (vi or gedit) and add the name of the new control file to the end of the list of control files. Include the path. Be certain not to enter spaces between the single quotes and commas in the control\_files= line. Be certain that this line is one continuous line, without line breaks. Save and close the file.

$ **vi $ORACLE\_HOME/dbs/initorclcdb.ora**

...

\*.control\_files='/u01/app/oracle/oradata/ORCLCDB/control01.ctl',

'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl',

**'/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl'**

...

$

1. Start SQL\*Plus and connect to the root container as the SYS user with the SYSDBA privilege. You are connected to an idle instance.

$ **sqlplus / AS SYSDBA**

…

Connected to an idle instance.

SQL>

1. Start the database instance.

SQL> **STARTUP**

ORACLE instance started.

…

Database mounted. Database opened.

SQL>

1. View the CONTROL\_FILES parameter again.

SQL> **SHOW PARAMETER control\_files**

NAME

TYPE

VALUE

control\_files

string

/u01/app/oracle/oradata/ORCLCDB/control01.ctl,

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl

SQL>

**Question:** Why does the CONTROL\_FILES parameter still show only two control files? **Answer:** By default, the database instance starts up with the SPFILE. If an SPFILE does not exist, then the instance starts up with a PFILE. In this case, both an SPFILE and PFILE

are present, so the SPFILE takes precedence. You configured the PFILE, not the SPFILE.

The SPFILE still contains only two references.

1. Re-create the third control file because the current version is no longer an exact copy of the others.
   1. Shut down the database instance with the IMMEDIATE option.

SQL> **SHUTDOWN IMMEDIATE**

Database closed. Database dismounted.

ORACLE instance shut down.

SQL>

* 1. Exit SQL\*Plus.

SQL> **EXIT**

* 1. Use the cp command to re-create control03.ctl.

$ **yes | cp –rf /u01/app/oracle/oradata/ORCLCDB/control01.ctl**

**/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl**

$

1. Re-create the SPFILE from the updated PFILE.
   1. Start SQL\*Plus and connect to the CDB root as the SYS user with the SYSDBA privilege. You are connected to an idle instance.

$ **sqlplus / AS SYSDBA**

…

Connected to an idle instance.

SQL>

* 1. Create the SPFILE.

SQL> **CREATE SPFILE FROM PFILE;**

File created.

SQL>

1. Start the database instance.

SQL> **STARTUP**

ORACLE instance started.

…

Database mounted. Database opened.

SQL>

1. View the CONTROL\_FILES parameter again. The third control file is now included in the list, which indicates that the SPFILE is configured properly. The results below are formatted for easier viewing.

SQL> **SHOW PARAMETER control\_files**

NAME

TYPE

VALUE

control\_files

string

/u01/app/oracle/oradata/ORCLCDB/control01.ctl,

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl,

/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl

SQL>

1. Query the V$CONTROLFILE view to confirm the number of control files. The result indicates that three control files are defined.

SQL> **SELECT name FROM v$controlfile;**

NAME

/u01/app/oracle/oradata/ORCLCDB/control01.ctl

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl

/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl

SQL>

**Practice 2-2: Configuring the Size of the Fast Recovery Area**

### Overview

In this practice, you review the fast recovery area (FRA) configuration and change its size to 12 GB.

### Assumptions

You are logged in to SQL\*Plus from the previous practice.

#### Time Estimate

It should take approximately 15 minutes to complete this practice.

### Tasks

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

View the values of the DB\_RECOVERY\_FILE\_DEST and DB\_RECOVERY\_FILE\_ DEST\_SIZE initialization parameters.

SQL> **SHOW PARAMETER db\_recovery\_file\_dest**

NAME

TYPE

VALUE

db\_recovery\_file\_dest

/u01/app/oracle/fast\_recovery\_area db\_recovery\_file\_dest\_size

14970M

SQL>

string

big integer

**Question:** Is the fast recovery area enabled?

**Answer:** Yes. The DB\_RECOVERY\_FILE\_DEST and DB\_RECOVERY\_FILE\_DEST\_SIZE

parameters values are not null, indicating that the fast recovery area is enabled.

**Question:** What changes can you make to the fast recovery area?

**Answer:** You can change the location and size of the fast recovery area.

**Question:** Does changing the size of the fast recovery area require the database to be restarted?

**Answer:** No, a restart is not required for this change because the

DB\_RECOVERY\_FILE\_DEST\_SIZE parameter is dynamic.

1. Change the size of the fast recovery area to 12GB and set the scope to BOTH.

SQL> **ALTER SYSTEM SET db\_recovery\_file\_dest\_size = 12G SCOPE=both;**

System altered.

SQL>

**Note:** If the archived redo log file destination fills up or cannot be written to, the database will halt. You would then need to remove archived redo log files from the archived redo log file destination so that the database could resume operations. This activity is covered in one of the next practices.

1. View the DB\_RECOVERY\_FILE\_DEST\_SIZE initialization parameter again. The result verifies that the size has been set to 12GB.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SQL> **SHOW PARAMETER db\_recovery\_file\_dest\_size** | | | | |
| NAME |  | TYPE |  | VALUE |
| db\_recovery\_file\_dest\_size SQL> |  | big integer |  | 12G |

**Practice 2-3: Verifying that the Redo Log File Is Multiplexed**

### Overview

Ensure that there are at least two redo log members in each group. If you are using file system storage, then each member should be distributed on separate disks or controllers so that no single equipment failure impacts an entire log group. The loss of an entire current log group is one of the most serious media failures because it can result in data loss. The loss of a single member of a multi-member log group does not affect database operation (other than causing an alert to be published in the alert log). One set of members should be stored in the FRA.

### Assumptions

You are logged in to SQL\*Plus from the previous practice.

#### Time Estimate

It should take approximately 30 minutes to complete this practice.

### Tasks

1. Query V$LOGFILE to determine the configuration (number of members) for each redo log group. The result shows that there are currently three log groups (1, 2, and 3) and only one member in each group.

SQL> **SELECT group#, status, member FROM v$logfile;**

SQL>

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| GROUP# |  | STATUS |  | MEMBER |
| 3 |  |  |  | /u01/app/oracle/oradata/ORCLCDB/redo03.log |
| 2 |  |  |  | /u01/app/oracle/oradata/ORCLCDB/redo02.log |
| 1 |  |  |  | /u01/app/oracle/oradata/ORCLCDB/redo01.log |

**Question:** Why is it recommended to have three groups when two would be sufficient?

**Answer:** The Oracle Database server treats the online redo log groups as a circular buffer in which to store transaction information, filling one group and then moving on to the next. After all groups have been written to, the Oracle Database server begins overwriting information in the first log group. If the database is configured in ARCHIVELOG mode, the LGWR cannot overwrite data in the first log group if it has not been archived.

**Question:** Can multiplexing redo logs impact database performance?

**Answer:** Multiplexing redo logs may heavily influence database performance because a commit cannot complete until the transaction information has been written to the logs by LGWR. You must place your redo log files on your fastest disks served by your fastest controllers. If possible, do not place any other database files on the same disks as your redo log files. Because only one group is written to at a given time, there is no performance impact in having members from several groups on the same disk.

1. Add another member to each redo log group. Name each member redo*nn*b.log, where

*nn* represents the group number.

SQL> **ALTER DATABASE ADD LOGFILE MEMBER**

**'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo01b.log' TO GROUP 1;**

Database altered.

SQL> **ALTER DATABASE ADD LOGFILE MEMBER**

**'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo02b.log' TO GROUP 2;**

Database altered.

SQL> **ALTER DATABASE ADD LOGFILE MEMBER**

**'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo03b.log' TO GROUP 3;**

Database altered. SQL>

1. Verify that the redo log files are now multiplexed. The query result shows that each group has two members, and therefore, the redo log files are multiplexed. Observe the INVALID status of the newly added redo log members. This status is expected because the new members have not yet been written to by LGWR. When a log switch occurs and the group containing the new member becomes CURRENT, the new member's status will change to null.

SQL> **SELECT group#, status, member FROM v$logfile ORDER BY 1, 3;**

GROUP# STATUS MEMBER

1 INVALID

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo01b.log

1

/u01/app/oracle/oradata/ORCLCDB/redo01.log

2 INVALID

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo02b.log

2

/u01/app/oracle/oradata/ORCLCDB/redo02.log

3 INVALID

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo03b.log

3

/u01/app/oracle/oradata/ORCLCDB/redo03.log

6 rows selected.

SQL>

1. Switch the log files and observe the changes.
   1. Find out which log group is the current log group. In this example, the query result shows that group 1 is the current group. Your current group may be different.

SQL> **SELECT group#, members, archived, status FROM v$log;**

SQL>

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GROUP# |  | MEMBERS |  | ARC |  | STATUS |
| 1 |  | 2 |  | NO |  | CURRENT |
| 2 |  | 2 |  | NO |  | INACTIVE |
| 3 |  | 2 |  | NO |  | INACTIVE |

* 1. Switch the log files three times.

SQL> **ALTER SYSTEM SWITCH LOGFILE;**

System altered.

SQL> **ALTER SYSTEM SWITCH LOGFILE;**

System altered.

SQL> **ALTER SYSTEM SWITCH LOGFILE;**

System altered.

SQL>

* 1. Query the V$LOGFILE view again. Notice that as a result of the log file switch, the new members' statuses have changed to null.

SQL> **SELECT group#, status, member FROM v$logfile ORDER BY 1, 3;**

GROUP# STATUS MEMBER

1

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo01b.log

1

/u01/app/oracle/oradata/ORCLCDB/redo01.log

2

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo02b.log

2

/u01/app/oracle/oradata/ORCLCDB/redo02.log

3

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo03b.log

3

/u01/app/oracle/oradata/ORCLCDB/redo03.log

6 rows selected.

SQL>

SQL> **SELECT group#, members, archived, status FROM v$log;**

SQL>

* 1. Query the V$LOG view again to learn which log group is now the current group. In this example, the results show that the LGWR is writing to group 1. Your group may be different. Your statuses may be different too. An INACTIVE status means the log group is no longer needed for database instance recovery.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GROUP# |  | MEMBERS |  | ARC |  | STATUS |
| 1 |  | 2 |  | NO |  | CURRENT |
| 2 |  | 2 |  | NO |  | INACTIVE |
| 3 |  | 2 |  | NO |  | INACTIVE |

* 1. Switch the log file.

SQL> **ALTER SYSTEM SWITCH LOGFILE;**

System altered.

SQL>

* 1. Query the V$LOG view again. The current group has changed to group 2, and the former current group's status is now ACTIVE. Your current group may be different. An ACTIVE status means that the log group is active, but it’s not the current log group. It is needed for crash recovery. It may be in use for block recovery.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GROUP# |  | MEMBERS |  | ARC |  | STATUS |
| 1 |  | 2 |  | NO |  | ACTIVE |
| 2 |  | 2 |  | NO |  | CURRENT |
| 3 |  | 2 |  | NO |  | INACTIVE |

* 1. Switch the log file again.

SQL> **ALTER SYSTEM SWITCH LOGFILE;**

System altered.

SQL>

SQL> **SELECT group#, members, archived, status FROM v$log;**

SQL>

* 1. Query the V$LOG view again. The current group has changed again to group 3, and the status of both the other groups is now ACTIVE. Your current group may be different.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GROUP# |  | MEMBERS |  | ARC |  | STATUS |
| 1 |  | 2 |  | NO |  | ACTIVE |
| 2 |  | 2 |  | NO |  | ACTIVE |
| 3 |  | 2 |  | NO |  | CURRENT |

**Question:** Can the LGWR background process write to only one member of the CURRENT

SQL> **SELECT group#, members, archived, status FROM v$log;**

SQL>

group in case the other members are missing or damaged?

**Answer:** Yes, it can. As long as there is one member left in the CURRENT group, LGWR can work.

1. To save space in your course practice environment, drop the redo log file members you created in step 4.
   1. Determine which redo log group is current. You cannot drop a member of the current group.

SQL> **SELECT group#, status FROM v$log;**

GROUP# STATUS

1. ACTIVE
2. ACTIVE
3. CURRENT

SQL>

* 1. Drop the member in the previous group and then perform a log switch. In this example, group 3 is current, so the command in this example drops a member in group 2.

SQL> **ALTER DATABASE DROP LOGFILE MEMBER**

**'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo02b.log';**

Database altered.

SQL> **alter system switch logfile;**

System altered.

SQL>

* 1. Drop the member in the next group and then perform a log switch.

SQL> **ALTER DATABASE DROP LOGFILE MEMBER**

**'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo03b.log';**

Database altered.

SQL> **alter system switch logfile;**

System altered.

SQL>

* 1. Drop the member in the final group and then perform a log switch.

SQL> **ALTER DATABASE DROP LOGFILE MEMBER**

**'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo01b.log';**

Database altered.

SQL> **alter system switch logfile;**

System altered.

SQL>

* 1. Verify that each group now has only one member.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| GROUP# |  | MEMBERS |  | ARC |  | STATUS |
| 1 |  | 1 |  | NO |  | ACTIVE |
| 2 |  | 1 |  | NO |  | ACTIVE |
| 3 |  | 1 |  | NO |  | CURRENT |

* 1. Exit from SQL\*Plus.

SQL> **exit**

…

$

SQL> **SELECT group#, members, archived, status FROM v$log;**

SQL>

* 1. Remove the physical files from the operating system.

$ **rm –rf /u01/app/oracle/fast\_recovery\_area/ORCLCDB/redo\*.log**

$

* 1. Verify that the redo log files have been removed.

$ **ls /u01/app/oracle/fast\_recovery\_area/ORCLCDB**

archivelog

$

control02.ctl

onlinelog

**Practice 2-4: Configuring ARCHIVELOG Mode**

### Overview

In this practice, you configure your database for ARCHIVELOG mode so that redo logs are archived.

### Assumptions

You are logged in as the oracle user.

#### Time Estimate

It should take approximately 10 minutes to complete this practice.

### Tasks

1. Log in to SQL\*Plus as the SYS user with the SYSDBA privilege.

$ **sqlplus / AS SYSDBA**

… SQL>

1. Issue the ARCHIVE LOG LIST command to determine whether the database is in

ARCHIVELOG mode.

SQL> **archive log list**

Database log mode No Archive Mode

Automatic archival Disabled

Archive destination USE\_DB\_RECOVERY\_FILE\_DEST Oldest online log sequence 25

Current log sequence 27

SQL>

1. You must put the database in MOUNT mode to enable archiving. Shut down the database instance, start it in MOUNT mode, and then enable archiving.
   1. Shut down the database instance.

SQL> **shutdown immediate**

Database closed. Database dismounted.

ORACLE instance shut down.

SQL>

* 1. Start the database instance and mount the database.

SQL> **startup mount**

ORACLE instance started.

…

Database mounted.

SQL>

* 1. Enable archiving.

SQL> **alter database archivelog;**

Database altered. SQL>

* 1. Verify that the database is now in ARCHIVELOG mode.

SQL> **archive log list**

Database log mode Archive Mode

Automatic archival Enabled

Archive destination USE\_DB\_RECOVERY\_FILE\_DEST Oldest online log sequence 25

Next log sequence to archive 27 Current log sequence 27

SQL>

* 1. Open the database.

SQL> **alter database open;**

Database altered. SQL>

1. Exit from SQL\*Plus.

SQL> **EXIT**

…

$

**Practices for Lesson 3: Using**

**Recovery Manager (RMAN)**

**Practices for Lesson 3: Overview**

### Overview

In these practices, you will verify the default backup destination, set the RMAN display data and time format, verify that control file autobackup is configured, and verify the backup retention policy.

**Practice 3-1: Configuring the Default Backup Destination**

### Overview

In this practice, you determine the default backup destination setting and create a backup in the default location.

### Assumptions

You have a terminal window open with the environment variables set for the ORCLCDB database instance.

#### Timing Estimate

It should take approximately 5 minutes to complete this practice.

### Tasks

1. The DB\_RECOVERY\_FILE\_DEST parameter specifies the default location for the fast recovery area. View the parameter value by logging in to SQL\*Plus and using the SHOW PARAMETER command.

$ **sqlplus / as sysdba**

…

SQL> **show parameter db\_recovery\_file\_dest**

NAME

TYPE

VALUE

db\_recovery\_file\_dest

/u01/app/oracle/fast\_recovery\_area

db\_recovery\_file\_dest\_size

string

big integer 12G

SQL> **exit**

$

1. Log in to RMAN, execute the BACKUP DATABASE command, and then exit RMAN.

$ **rman target "'/ as sysbackup'"**

…

RMAN> **BACKUP DATABASE;**

Starting backup at 18-JUN-19

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

…

Finished backup at 18-JUN-19

…

Finished Control File and SPFILE Autobackup at 18-JUN-19 RMAN> **exit**

**Practice 3-2: Setting the Date and Time Format for RMAN**

### Overview

In this practice, you set the date/time format that RMAN uses for displaying time stamps. The default date format does not include the time of day information. You will be performing several backups and recoveries in the same day, and in some cases, you will need to determine which backup to use.

### Assumptions

You completed Practice 3-1.

You have a terminal window open with the environment variables set for the ORCLCDB database instance.

#### Timing Estimate

It should take approximately 10 minutes to complete this practice.

### Tasks

1. Set the NLS\_LANG and NLS\_DATE\_FORMAT variables so that RMAN includes time information in any time stamp values.

Use the editor of your choice (vi, gedit) and add the following two lines to the end of the

~/.bashrc file:

export NLS\_LANG=american\_america.al32utf8 export NLS\_DATE\_FORMAT="yyyy-mm-dd:hh24:mi:ss"

* 1. Run following command in the terminal as root user: **xhost +**
  2. In a terminal window, invoke your editor. This example uses the gedit editor.

$ **gedit ~/.bashrc**

* 1. Copy and paste by using the mouse buttons and icons in the editor, save your changes, and then close the gedit window.

**Note:** If you are not copying and pasting the variables, double-check that the

NLS\_LANG value is written correctly with a letter “l”, not a digit “1”.

* 1. View the modified file by using the cat command and apply the new settings.

$ **cat ~/.bashrc**

# .bashrc

…

export NLS\_LANG=american\_america.al32utf8 export NLS\_DATE\_FORMAT="yyyy-mm-dd:hh24:mi:ss"

$ **. .bashrc**

$

1. Verify the settings by starting RMAN and listing the backups in the recovery catalog or from the control file.
   1. Log in to RMAN.

$ **rman target "'/ as sysbackup'"**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

a. List the backups and note the time stamp format. The output shows the most recent backups.

RMAN> **list backup;**

using target database control file instead of recovery catalog

List of Backup Sets

===================

BS Key Time

Type LV Size

Device Type Elapsed Time Completion

1

Full

1.37G

DISK

00:00:11

2019-06-

18:17:21:59

BP Key: 1 Status: AVAILABLE TAG20190618T172148

Piece Name:

Compressed: NO

Tag:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019\_06\_18/ o1\_mf\_nnndf\_TAG20190618T172148\_gjl7fdf3\_.bkp

List of Datafiles in backup set 1

File LV Type Ckp SCN Name

Ckp Time

Abs Fuz SCN Sparse

1

Full 3133091

2019-06-18:17:21:48

NO

/u01/app/oracle/oradata/ORCLCDB/system01.dbf

…

BS Key Time

Type LV Size

Device Type Elapsed Time Completion

5

Full

17.95M

DISK

00:00:01

2019-06-

18:17:22:59

BP Key: 5 TAG20190618T172258

Status: AVAILABLE

Compressed: NO Tag:

Piece Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_06\_18

/o1\_mf\_s\_1011288178\_gjl7hm7n\_.bkp

SPFILE Included: Modification time: 2019-06-18:14:56:57 SPFILE db\_unique\_name: ORCLCDB

Control File Included: Ckp SCN: 3133134 Ckp time: 2019- 06-18:17:22:58

RMAN>

1. **Tip:** Because the output of the RMAN commands can be quite long, consider using the RMAN SPOOL LOG TO <file> command to direct the output to your specified file.
   1. Optionally, enter SPOOL LOG TO /tmp/test.log in your RMAN window.
   2. Repeat the previous LIST BACKUP command.
   3. Note that the output is not displayed in your window. In a second terminal window, enter cat /tmp/test.log to view the output.
   4. Use SPOOL LOG OFF to stop directing output to the file.

**Practice 3-3: Configuring RMAN Settings**

### Overview

In this practice, you configure or confirm RMAN settings for the following tasks:

* To back up the control file and server parameter file (SPFILE) each time RMAN takes a backup of anything in the orclcdb database
* To ensure that one redundant backup is kept

Then you perform a tablespace backup to verify these settings.

### Assumptions

You completed Practice 3-2.

You have a terminal window open with the environment variables set for the ORCLCDB database instance.

#### Timing Estimate

It should take approximately 5 minutes to complete this practice.

### Tasks

1. While still logged in to an RMAN session as SYSBACKUP, verify that control file autobackup is enabled.

RMAN> **show controlfile autobackup;**

RMAN configuration parameters for database with db\_unique\_name ORCLCDB are:

CONFIGURE CONTROLFILE AUTOBACKUP ON; # default

RMAN>

1. View the current retention policy setting to verify that it is set to 1.

RMAN> **show retention policy;**

RMAN configuration parameters for database with db\_unique\_name ORCLCDB are:

CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default

RMAN>

1. Determine which data file belongs to the USERS tablespace in ORCLPDB1.

RMAN> **REPORT SCHEMA;**

Report of database schema for database with db\_unique\_name ORCLCDB

List of Permanent Datafiles

===========================

File Size(MB) Tablespace

RB segs Datafile Name

1

920

SYSTEM

YES

/u01/app/oracle/oradata/ORCLCDB/system01.dbf

3 850 SYSAUX NO

/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf

…

12 342 ORCLPDB1:USERS NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

…

4 36 ORCLPDB2:TEMP 32767

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/temp01.dbf

RMAN>

1. Back up the USERS tablespace in both the CDB and ORCLPDB1.

RMAN> **BACKUP TABLESPACE users, ORCLPDB1:USERS;**

Starting backup at 2019-06-18:18:04:29 using channel ORA\_DISK\_1

channel ORA\_DISK\_1: starting full datafile backup set channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00012 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

…

Finished backup at 2019-06-18:18:04:31

Starting Control File and SPFILE Autobackup at 2019-06- 18:18:04:31

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_18/o1\_mf\_s\_1011290671\_gjl9xhwf\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 18:18:04:32

RMAN>

1. Exit RMAN.

RMAN> **exit**

Recovery Manager complete.

$

1. Close the terminal window opened for this practice.

**Practices for Lesson 4:**

**Backup Strategies**

**Practices for Lesson 4**

There are no practices for Lesson 4.

**Practices for Lesson 5:**

**Creating Database Backups**

**Practices for Lesson 5: Overview**

### Overview

In these practices, you will create a script file that can be used to re-create the control file. You will also create a whole database backup and a partial database backup.

**Practice 5-1: Backing Up the Control File**

### Overview

In this practice, you back up your control file to a trace file and then create a file of SQL commands that can be used to re-create the control file.

#### Tip

The loss of a single control file causes the database instance to fail because all control files must be available at all times. However, recovery can be a simple matter of copying one of the other control files. The loss of all control files is slightly more difficult to recover from, but is not usually catastrophic as long as you created a copy of the control file by backing it up to a trace file.

### Assumptions

You are logged in as the oracle user.

#### Timing Estimate

It should take approximately 20 minutes to complete this practice.

### Tasks

1. Open a terminal window and use the oraenv script to set the environment variables for the

ORCLCDB database.

$ **. oraenv**

ORACLE\_SID = [orclcdb] ?

The Oracle base remains unchanged with value /u01/app/oracle

$

1. Make sure the database and listener is running.

$ **lsnrctl status**

1. Start SQL\*Plus and connect to the CDB root as the SYS user with SYSDBA privileges.

$ **sqlplus / as sysdba**

…

SQL>

1. Verify that the control files are multiplexed.

SQL> **SELECT name FROM v$controlfile;**

NAME

/u01/app/oracle/oradata/ORCLCDB/control01.ctl

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl

/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl

SQL>

1. Back up the control file to a trace file.

SQL> **ALTER DATABASE BACKUP controlfile TO trace;**

Database altered. SQL>

1. Exit SQL\*Plus.

SQL> **EXIT**

…

$

1. Navigate to the directory that contains the alert log file and trace files.

$ **cd /u01/app/oracle/diag/rdbms/orclcdb/orclcdb/trace**

[trace]$

1. List the files in this directory. Notice that the directory contains the alert log (alert\_orclcdb.log) and many trace files (.trc).

[trace]$ **ls**

alert\_orclcdb.log orclcdb\_ora\_8708.trc orclcdb\_dia0\_12061\_base\_1.trc orclcdb\_ora\_8708.trm

…

[trace]$

1. View the end of the alert log and make note of the last trace file created as a backup for the control file. In this example, it is orclcdb\_ora\_8708.trc. Your file name will be different.

[trace]$ **tail alert\_orclcdb.log**

…

alter database backup controlfile to trace 2019-06-20T13:24:51.885054+00:00

Backup controlfile written to trace file

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/trace/orclcdb\_ora\_870 8.trc

Completed: alter database backup controlfile to trace

…

[trace]$

1. View the content of the last generated trace file by using the cat command. Make sure to substitute the name of the trace file with your trace file name.

Between the lines " -- Set #1. NORESETLOGS case" and " -- Set #2. RESETLOGS case", select the code from STARTUP NOMOUNT to ALTER SESSION SET CONTAINER = CDB$ROOT; and copy it to the clipboard.

**Note:** The file names in your database will likely differ from the file names shown in this example.

[trace]$ **gedit orclcdb\_ora\_UPDATE\_FILE\_NUMBER.trc**

…

STARTUP NOMOUNT

CREATE CONTROLFILE REUSE DATABASE "ORCLCDB" NORESETLOGS ARCHIVELOG

MAXLOGFILES 16

MAXLOGMEMBERS 3

MAXDATAFILES 1024

MAXINSTANCES 8

MAXLOGHISTORY 292 LOGFILE

GROUP 1 '/u01/app/oracle/oradata/ORCLCDB/redo01.log' SIZE 200M BLOCKSIZE 512,

GROUP 2 '/u01/app/oracle/oradata/ORCLCDB/redo02.log' SIZE 200M BLOCKSIZE 512,

GROUP 3 '/u01/app/oracle/oradata/ORCLCDB/redo03.log' SIZE 200M BLOCKSIZE 512

-- STANDBY LOGFILE DATAFILE

'/u01/app/oracle/oradata/ORCLCDB/system01.dbf', '/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf', '/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf', '/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf', '/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf', '/u01/app/oracle/oradata/ORCLCDB/users01.dbf', '/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf', '/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf'

CHARACTER SET AL32UTF8

;

-- Commands to re-create incarnation table

-- Below log names MUST be changed to existing filenames on

-- disk. Any one log file from each branch can be used to

-- re-create incarnation records.

-- ALTER DATABASE REGISTER LOGFILE

'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_2 0/o1\_mf\_1\_1\_%u\_.arc';

-- ALTER DATABASE REGISTER LOGFILE

'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_2 0/o1\_mf\_1\_1\_%u\_.arc';

-- Recovery is required if any of the datafiles are restored backups,

-- or if the last shutdown was not normal or immediate. RECOVER DATABASE

-- All logs need archiving and a log switch is needed. ALTER SYSTEM ARCHIVE LOG ALL;

-- Database can now be opened normally. ALTER DATABASE OPEN;

-- Open all the PDBs.

ALTER PLUGGABLE DATABASE ALL OPEN;

-- Commands to add tempfiles to temporary tablespaces.

-- Online tempfiles have complete space information.

-- Other tempfiles may require adjustment. ALTER TABLESPACE TEMP ADD TEMPFILE

'/u01/app/oracle/oradata/ORCLCDB/temp01.dbf'

SIZE 137363456 REUSE AUTOEXTEND ON NEXT 655360 MAXSIZE 32767M;

ALTER SESSION SET CONTAINER = "PDB$SEED"; ALTER TABLESPACE TEMP ADD TEMPFILE

'/u01/app/oracle/oradata/ORCLCDB/pdbseed/temp012019-05-08\_01-48- 43-006-AM.dbf'

SIZE 37748736 REUSE AUTOEXTEND ON NEXT 655360 MAXSIZE 32767M;

ALTER SESSION SET CONTAINER = "ORCLPDB1"; ALTER TABLESPACE TEMP ADD TEMPFILE

'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/temp01.dbf'

SIZE 37748736 REUSE AUTOEXTEND ON NEXT 655360 MAXSIZE 32767M;

ALTER SESSION SET CONTAINER = "ORCLPDB2";

ALTER TABLESPACE TEMP ADD TEMPFILE

'/u01/app/oracle/oradata/ORCLCDB/orclpdb2/temp01.dbf'

SIZE 37748736 REUSE AUTOEXTEND ON NEXT 655360 MAXSIZE 32767M;

ALTER SESSION SET CONTAINER = "CDB$ROOT";

…

[trace]$

1. Open an editor and paste the code you copied in the previous step into a new file named

**ControlFileBackup.sql** in the /home/oracle directory and then save the file.

[trace]$ **cd $HOME**

$ **vi ControlFileBackup.sql**

*Paste the code from the previous step into the file.*

$

**Question:** Which command would allow the re-creation of the control files in case of a complete loss of the control files?

**Answer:** In the case where all control files are lost, the CREATE CONTROLFILE command in the trace file would re-create the missing control files with the right information, keeping the database file structure in terms of data files, redo log files, and other database attributes (ARCHIVELOG, maximum settings).

**Question:** How would you execute the command?

**Answer:** After trimming the trace file by keeping all commands from the STARTUP NOMOUNT up to ALTER SESSION SET CONTAINER = CDB$ROOT;, you would execute the file as a SQL script.

**Question:** Are the data files, temp files, and control files that structure the ORCLCDB

database included in the SQL script?

**Answer:** Yes, they are included. All data and temp files of the different containers (the CDB root, CDB seed, PDB1, and so on) and the multiplexed redo log files are present.

**Question:** Which other attributes structure the ORCLCDB database?

**Answer:** The ARCHIVELOG mode, the character set, and the name of the CDB

**Question:** Why are there two cases—Set #1. NORESETLOGS and Set #2. RESETLOGS?

**Answer:** The first case from the STARTUP NOMOUNT to the ALTER SESSION SET CONTAINER = CDB$ROOT provides a script to execute a complete database recovery. Use this only if the current versions of all online logs are available. The second case provides a script to execute an incomplete database recovery. Use this only if online logs are damaged. The contents of online logs will be lost, and all backups will be invalidated.

**Question:** When would you have to regenerate the trace file from the current control files? **Answer:** Because the control file changes after each data file or redo log file change (adding, removing, resizing) or database attribute change (ARCHIVELOG), you would have

to redo the backup of your control file to a trace file.

**Practice 5-2: Verifying Automatic Backups of the Control File and SPFILE**

### Overview

In this practice, you use Recovery Manager (RMAN) to configure automatic backups of the control file and server parameter file (SPFILE) when a backup of the database is made and when there is a structural change to the database.

### Assumptions

You are logged in to the compute node as the oracle user.

#### Timing Estimate

It should take approximately 10 minutes to complete this practice.

### Tasks

1. Start Recovery Manager and connect to the CDB root (target database) as the SYS user.

$ **rman target /**

**…**

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. Show all RMAN settings. Notice the CONFIGURE CONTROLFILE AUTOBACKUP ON;

setting.

RMAN> **SHOW ALL;**

using target database control file instead of recovery catalog

RMAN configuration parameters for database with db\_unique\_name ORCLCDB are:

CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default CONFIGURE BACKUP OPTIMIZATION OFF; # default CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default CONFIGURE CONTROLFILE AUTOBACKUP ON; # default

CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO

'%F'; # default

CONFIGURE DEVICE TYPE DISK PARALLELISM 1 BACKUP TYPE TO

BACKUPSET; # default

CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE DISK TO 1; #

default

CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE DISK TO 1; #

default

CONFIGURE MAXSETSIZE TO UNLIMITED; # default CONFIGURE ENCRYPTION FOR DATABASE OFF; # default CONFIGURE ENCRYPTION ALGORITHM 'AES128'; # default

CONFIGURE COMPRESSION ALGORITHM 'BASIC' AS OF RELEASE 'DEFAULT' OPTIMIZE FOR LOAD TRUE ; # default

CONFIGURE RMAN OUTPUT TO KEEP FOR 7 DAYS; # default

CONFIGURE ARCHIVELOG DELETION POLICY TO NONE; # default CONFIGURE SNAPSHOT CONTROLFILE NAME TO

'/u01/app/oracle/product/19.3.0/dbhome\_1/dbs/snapcf\_orclcdb.f'; # default

RMAN>

**Question:** In your configuration, does RMAN automatically back up the control file and server parameter file (SPFILE) with every backup and database structural change?

**Answer:** Yes, it does because the CONTROLFILE AUTOBACKUP attribute is set to ON. **Question:** Will a backup operation back up all control files or only one of the multiplexed control files?

**Answer:** It will back up only one of the multiplexed control files because all control files in a database are identical.

1. Exit RMAN.

RMAN> **EXIT**

Recovery Manager complete.

$

**Practice 5-3: Creating a Whole Database Backup**

### Overview

In this practice, you use Recovery Manager to back up your entire database, including the archived redo log files, the SPFILE, and the control files. The backup should be the base for an incremental backup strategy.

### Assumptions

You are logged in as the oracle user.

#### Timing Estimate

It should take approximately 30 minutes to complete this practice.

### Tasks

1. Start Oracle Recovery Manager (RMAN) and connect to the CDB root as the SYS user.

$ **rman target /**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. View the structure of the CDB in terms of PDBs, tablespaces, and data files (permanent and temporary). Your file numbers will differ from those shown below.

RMAN> **REPORT schema;**

Report of database schema for database with db\_unique\_name ORCLCDB

List of Permanent Datafiles

===========================

File Size(MB) Tablespace

RB segs Datafile Name

1

920

SYSTEM

YES

/u01/app/oracle/oradata/ORCLCDB/system01.dbf

1. 850 SYSAUX NO

/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf

1. 330 UNDOTBS1 YES

/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf

1. 270 PDB$SEED:SYSTEM NO

/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf

1. 330 PDB$SEED:SYSAUX NO

/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf

1. 5 USERS NO

/u01/app/oracle/oradata/ORCLCDB/users01.dbf

* 1. 100 PDB$SEED:UNDOTBS1 NO

/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf

* 1. 280 ORCLPDB1:SYSTEM YES

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf

* 1. 370 ORCLPDB1:SYSAUX NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf

* 1. 100 ORCLPDB1:UNDOTBS1 YES

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf

* 1. 342 ORCLPDB1:USERS NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

* 1. 280 ORCLPDB2:SYSTEM YES

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf

* 1. 370 ORCLPDB2:SYSAUX NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf

* 1. 100 ORCLPDB2:UNDOTBS1 YES

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf

* 1. 342 ORCLPDB2:USERS NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf

List of Temporary Files

=======================

File Size(MB) Tablespace Maxsize(MB) Tempfile Name

1 131 TEMP 32767

/u01/app/oracle/oradata/ORCLCDB/temp01.dbf

1. 36 PDB$SEED:TEMP 32767

/u01/app/oracle/oradata/ORCLCDB/pdbseed/temp012019-05-08\_01-48- 43-006-AM.dbf

1. 36 ORCLPDB1:TEMP 32767

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/temp01.dbf

1. 36 ORCLPDB2:TEMP 32767

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/temp01.dbf

RMAN>

1. Back up the whole database. Your results will be different from the results shown below; for example, the piece handle names will be different.

RMAN> **BACKUP DATABASE;**

Starting backup at 2019-06-20:15:46:55 allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=39 device type=DISK

channel ORA\_DISK\_1: starting full datafile backup set channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00001 name=/u01/app/oracle/oradata/ORCLCDB/system01.dbf

input datafile file number=00003 name=/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf

input datafile file number=00004 name=/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf

input datafile file number=00007 name=/u01/app/oracle/oradata/ORCLCDB/users01.dbf

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:15:46:56

channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:15:47:11

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_20/o1\_mf\_nnndf\_TAG20190620T154656\_gjqbmjks\_.bkp tag=TAG20190620T154656 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:15 channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00010 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf

input datafile file number=00012 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

input datafile file number=00009 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf

input datafile file number=00011 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:15:47:11 channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:15:47:18

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B36632797E 5CE0536210ED0ADAC7/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T 154656\_gjqbmzl3\_.bkp tag=TAG20190620T154656 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00014 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf

input datafile file number=00016 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf

input datafile file number=00013 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf

input datafile file number=00015 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:15:47:18 channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:15:47:25

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B419BF707E 73E0536210ED0A54C7/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T 154656\_gjqbn6qg\_.bkp tag=TAG20190620T154656 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00006 name=/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf

input datafile file number=00005 name=/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf

input datafile file number=00008 name=/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:15:47:25 channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:15:47:32

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857888C961176 10E0536210ED0A12EE/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T 154656\_gjqbnfv8\_.bkp tag=TAG20190620T154656 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 Finished backup at 2019-06-20:15:47:32

Starting Control File and SPFILE Autobackup at 2019-06- 20:15:47:32

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011455252\_gjqbno6q\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:15:47:33

RMAN>

**Question:** Do you have to shut down the database to back it up?

**Answer:** No, as long as the database is in ARCHIVELOG mode, the backup can take place while the database is opened. This is a hot backup (or online backup). A cold backup (or offline backup) is a backup completed while the database is closed and is required if the database is in NOARCHIVELOG mode.

**Question:** Are hot backups consistent?

**Answer:** Online backups are inconsistent because with the database opened, there is no guarantee that the data files are synchronized with the control files. However, offline backups taken while the database is not opened are consistent because, at the time of the backup, the system change number (SCN) in data file headers matches the SCN in the control files.

**Question:** How can hot backups (inconsistent backups) be used in complete database recovery?

**Answer:** During a complete recovery, restored online backups are recovered until the current SCN is matched, with the use of the archive log files and online redo log files.

**Question:** Did the backup include the SPFILE and control files?

**Answer:** Yes. This is the last operation completed at the end of the backup command.

…

Starting Control File and SPFILE Autobackup at 2019-06- 20:15:47:32

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011455252\_gjqbno6q\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:15:47:33

RMAN>

**Question:** Does the complete operation create a single backup set?

**Answer:** No. The operation creates multiple backup sets.

* Four backup sets including data files (one for each of the containers): CDB root, PDB seed, ORCLPDB1, and ORCLPDB2
* One backup set for the SPFILE and control files.

1. List the backup sets. Look for Piece Name in the results for each backup set.

RMAN> **LIST BACKUP;**

List of Backup Sets

===================

…

20:15:46:59

BP Key: 18 TAG20190620T154656

Piece Name:

Status: AVAILABLE

Compressed: NO Tag:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019\_06\_20/ o1\_mf\_nnndf\_TAG20190620T154656\_gjqbmjks\_.bkp

List of Datafiles in backup set 18

File LV Type Ckp SCN Name

Ckp Time

Abs Fuz SCN Sparse

1

Full 3357562

2019-06-20:15:46:56

NO

/u01/app/oracle/oradata/ORCLCDB/system01.dbf

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BS Key Time |  | Type LV |  | Size |  | Device Type |  | Elapsed Time |  | Completion |
| 18 |  | Full |  | 1.40G |  | DISK |  | 00:00:03 |  | 2019-06- |

3 Full 3357562 2019-06-20:15:46:56 NO

/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf

4 Full 3357562 2019-06-20:15:46:56 NO

/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf

7 Full 3357562 2019-06-20:15:46:56 NO

/u01/app/oracle/oradata/ORCLCDB/users01.dbf

BS Key Type LV Size Device Type Elapsed Time Completion Time

19 Full 574.40M DISK 00:00:01 2019-06-

20:15:47:12

BP Key: 19 Status: AVAILABLE Compressed: NO Tag: TAG20190620T154656

Piece Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B36632797E5CE0536 210ED0ADAC7/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T154656\_ gjqbmzl3\_.bkp

List of Datafiles in backup set 19 Container ID: 3, PDB Name: ORCLPDB1

File LV Type Ckp SCN Ckp Time Abs Fuz SCN Sparse Name

9 Full 3357568 2019-06-20:15:47:11 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf

10 Full 3357568 2019-06-20:15:47:11 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf

11 Full 3357568 2019-06-20:15:47:11 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf

12 Full 3357568 2019-06-20:15:47:11 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BS Key Time |  | Type LV |  | Size |  | Device Type |  | Elapsed Time |  | Completion |
| 20 |  | Full |  | 574.20M |  | DISK |  | 00:00:01 |  | 2019-06- |

20:15:47:19

BP Key: 20 Status: AVAILABLE Compressed: NO Tag: TAG20190620T154656

Piece Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B419BF707E73E0536 210ED0A54C7/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T154656\_ gjqbn6qg\_.bkp

List of Datafiles in backup set 20 Container ID: 4, PDB Name: ORCLPDB2

File LV Type Ckp SCN Ckp Time Abs Fuz SCN Sparse Name

13 Full 3357571 2019-06-20:15:47:18 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf

14 Full 3357571 2019-06-20:15:47:18 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf

15 Full 3357571 2019-06-20:15:47:18 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf

16 Full 3357571 2019-06-20:15:47:18 NO

/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BS Key Time |  | Type LV |  | Size |  | Device Type |  | Elapsed Time |  | Completion |
| 21 |  | Full |  | 554.96M |  | DISK |  | 00:00:01 |  | 2019-06- |

20:15:47:26

BP Key: 21 Status: AVAILABLE Compressed: NO Tag: TAG20190620T154656

Piece Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857888C96117610E0536 210ED0A12EE/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T154656\_ gjqbnfv8\_.bkp

List of Datafiles in backup set 21 Container ID: 2, PDB Name: PDB$SEED

File LV Type Ckp SCN Ckp Time Abs Fuz SCN Sparse Name

5 Full 2158685 2019-05-08:01:59:54 NO

/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf

6 Full 2158685 2019-05-08:01:59:54 NO

/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf

8 Full 2158685 2019-05-08:01:59:54 NO

/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| BS Key Time |  | Type LV |  | Size |  | Device Type |  | Elapsed Time |  | Completion |
| 22 |  | Full |  | 17.95M |  | DISK |  | 00:00:01 |  | 2019-06- |

20:15:47:33

BP Key: 22 Status: AVAILABLE Compressed: NO Tag: TAG20190620T154732

Piece Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_06\_20

/o1\_mf\_s\_1011455252\_gjqbno6q\_.bkp

SPFILE Included: Modification time: 2019-06-20:05:00:49

SPFILE db\_unique\_name: ORCLCDB

Control File Included: Ckp SCN: 3357584 06-20:15:47:32

Ckp time: 2019-

RMAN>

1. Exit RMAN.

RMAN> **EXIT**

1. Verify that the files are stored on disk in the FRA.

#### $ cd /u01/app/oracle/fast\_recovery\_area/ORCLCDB

$ **ls -ltR**

…

./8857B36632797E5CE0536210ED0ADAC7:

total 4

drwxr-x--- 4 oracle oinstall 4096 Jun 20 15:47 backupset

./8857B36632797E5CE0536210ED0ADAC7/backupset:

total 8

drwxr-x--- 2 oracle oinstall 4096 Jun 20 15:47 2019\_06\_20

drwxr-x--- 2 oracle oinstall 4096 Jun 18 19:39 2019\_06\_18

./8857B36632797E5CE0536210ED0ADAC7/backupset/2019\_06\_20:

total 588196

-rw-r----- 1 oracle oinstall 602308608 Jun 20 15:47 o1\_mf\_nnndf\_TAG20190620T154656\_gjqbmzl3\_.bkp

./8857B36632797E5CE0536210ED0ADAC7/backupset/2019\_06\_18:

total 1305840

-rw-r----- 1 oracle oinstall 66494464 Jun 18 19:39 o1\_mf\_nnndf\_TAG20190618T193903\_gjlhgq5r\_.bkp

-rw-r----- 1 oracle oinstall 602128384 Jun 18 19:29 o1\_mf\_nnndf\_TAG20190618T192926\_gjlgx5j1\_.bkp

-rw-r----- 1 oracle oinstall 66494464 Jun 18 18:04 o1\_mf\_nnndf\_TAG20190618T180429\_gjl9xfgb\_.bkp

-rw-r----- 1 oracle oinstall 602054656 Jun 18 17:22 o1\_mf\_nnndf\_TAG20190618T172148\_gjl7g5kp\_.bkp

./onlinelog:

total 0

$

**Question:** Where are the backups of control files and SPFILE located?

**Answer:** They are created in the autobackup subdirectory.

**Question:** How are backups deleted?

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

1. View the backup retention policy.
2. Start RMAN and connect to the CDB root as the SYS user.

$ **rman target /**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. Issue the SHOW RETENTION POLICY command. The policy is REDUNDANCY 1.

RMAN> **SHOW RETENTION POLICY;**

using target database control file instead of recovery catalog

RMAN configuration parameters for database with db\_unique\_name ORCLCDB are:

CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default

RMAN>

**Question:** How does Oracle determine when files are obsolete?

**Answer:** There are two retention policy parameters that are mutually exclusive:

* If a retention policy is enabled with RECOVERY WINDOW OF 5 DAYS, the window stretches from the current time (SYSDATE) to the point of recoverability, which is the earliest date to which you want to recover. The point of recoverability is SYSDATE - integer days in the past.
* If a retention policy is enabled with REDUNDANCY r, then RMAN skips backups only if at least *n* backups of an identical file exist on the specified device, where n=r+1 (default is 1).

RMAN automatically deletes obsolete backup sets and copies in the FRA when space is needed.

1. Manually delete obsolete files by issuing the DELETE OBSOLETE command. If there are obsolete files, respond YES when prompted. Your results will differ from this example.

RMAN> **delete obsolete;**

RMAN retention policy will be applied to the command RMAN retention policy is set to redundancy 1 allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=32 device type=DISK Deleting the following obsolete backups and copies:

Type Key Completion Time Filename/Handle

Backup Set

2

2019-06-18:17:22:22

Backup Piece

2

2019-06-18:17:22:22

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B36632797E5CE0536 210ED0ADAC7/backupset/2019\_06\_18/o1\_mf\_nnndf\_TAG20190618T172148\_ gjl7g5kp\_.bkp

Backup Set

…

3

2019-06-18:17:22:37

Do you really want to delete the above objects (enter YES or NO)? YES

deleted backup piece

backup piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B36632797E 5CE0536210ED0ADAC7/backupset/2019\_06\_18/o1\_mf\_nnndf\_TAG20190618T

172148\_gjl7g5kp\_.bkp RECID=2 STAMP=1011288133

…

deleted archived log

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_20/o1\_mf\_1\_35\_gjq0tnsk\_.arc RECID=9 STAMP=1011445205

Deleted 25 objects

RMAN>

1. Back up the database and archive logs as image copies. At the same time, free space in the FRA by deleting the archive log files once they are backed up.

a. Perform the backup (it will take few minutes) .

RMAN> **BACKUP AS COPY DATABASE PLUS ARCHIVELOG DELETE INPUT;**

Starting backup at 2019-06-20:16:05:18 current log archived

using channel ORA\_DISK\_1

channel ORA\_DISK\_1: starting archived log copy

input archived log thread=1 sequence=36 RECID=10 STAMP=1011456318

output file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_20/o1\_mf\_1\_36\_gjqcoyl6\_.arc RECID=11 STAMP=1011456318

channel ORA\_DISK\_1: archived log copy complete, elapsed time: 00:00:01

channel ORA\_DISK\_1: deleting archived log(s) archived log file

name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_20/o1\_mf\_1\_36\_gjqcoy6m\_.arc RECID=10 STAMP=1011456318

Finished backup at 2019-06-20:16:05:19

…

Starting Control File and SPFILE Autobackup at 2019-06- 20:16:06:27

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011456388\_gjqcr4bo\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:16:06:29

RMAN>

**Question:** What would you do if an error such as the following occurs?

RMAN-00571:

===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS

===============

RMAN-00571:

===========================================================

RMAN-03002: failure of backup plus archivelog command at 06/20/2019 11:05:08

ORA-19809: limit exceeded for recovery files

ORA-19804: cannot reclaim 67108864 bytes disk space from 19327352832 bytes limit

**Answer:** Increase the DB\_RECOVERY\_FILE\_DEST\_SIZE parameter value to 30G by issuing the following command:

RMAN> ALTER SYSTEM SET db\_recovery\_file\_dest\_size = 30G SCOPE=both;

**Question:** What is the advantage of creating backups as image copies?

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

**Question:** What is the advantage of creating backups as backup sets?

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

**Question:** How many image copies of the data files are created?

**Answer:** There are 15 image copies, one image copy for each data file in the CDB, PDBs included.

1. Exit RMAN.

RMAN> **EXIT**

**Practice 5-4: Creating Partial Database Backups**

### Overview

In this practice, you use Recovery Manager to back up ORCLPDB1, including the archived redo log files. You also back up a specific tablespace in ORCLPDB2.

### Assumptions

You are logged in as the oracle user.

#### Time Estimate

It should take approximately 20 minutes to complete this practice.

### Tasks

1. Start Recovery Manager (RMAN) and connect to the CDB root as the SYS user.

$ **rman target /**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. Back up ORCLPDB1, including the archived redo log files.

RMAN> **BACKUP PLUGGABLE DATABASE ORCLPDB1 PLUS ARCHIVELOG;**

Starting backup at 2019-06-20:17:57:57 current log archived

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=37 device type=DISK

channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set

input archived log thread=1 sequence=36 RECID=11 STAMP=1011456318

input archived log thread=1 sequence=37 RECID=13 STAMP=1011456386

input archived log thread=1 sequence=38 RECID=14 STAMP=1011463077

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:17:57:58 channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:17:57:59

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_20/o1\_mf\_annnn\_TAG20190620T175758\_gjql96rc\_.bkp tag=TAG20190620T175758 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:01 Finished backup at 2019-06-20:17:57:59

…

Starting Control File and SPFILE Autobackup at 2019-06- 20:17:58:08

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011463088\_gjql9jrr\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:17:58:09

RMAN>

1. Exit RMAN.

RMAN> **EXIT**

…

$

**Question:** Did the partial backup automatically include the SPFILE and control files?

**Answer:** Yes. Autobackup is also valid for partial backups.

**Question:** How many backup sets are created?

**Answer:** Four backup sets: one for the PDB data files, one for the SPFILE and control file, one for the archived log files before the data file backup set, and one for the archived log files after the data file backup set.

**Question:** In RMAN, can you connect directly to the PDB to perform the same backup?

**Answer:** Yes. In this case, you do not have to specify that you want to back up a PDB. Instead, you can use the BACKUP DATABASE command.

1. Perform a partial database backup in ORCLPDB1 directly.
   1. Start RMAN and connect to ORCLPDB1 as the SYS user.

$ **export** **ORACLE\_PDB\_SID=ORCLPDB1**

$ **rman target SYS/fenago**

**…**

connected to target database: ORCLCDB:ORCLPDB1 (DBID=2991365572) RMAN>

* 1. Execute the BACKUP DATABASE command. Notice that the SPFILE and control file are not backed up.

RMAN> **BACKUP DATABASE;**

Starting backup at 2019-06-20:18:14:59

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=32 device type=DISK

channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00010 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf

input datafile file number=00012 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

input datafile file number=00009 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf

input datafile file number=00011 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:18:15:00 channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:18:15:07

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B36632797E 5CE0536210ED0ADAC7/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T 181500\_gjqm94sz\_.bkp tag=TAG20190620T181500 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 Finished backup at 2019-06-20:18:15:07

RMAN>

1. Try to configure the recovery setting for the PDB so that the SPFILE and control file are backed up too.

RMAN> **CONFIGURE CONTROLFILE AUTOBACKUP ON;**

RMAN-00571:

===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS

===============

RMAN-00571:

===========================================================

RMAN-03002: failure of configure command at 06/20/2019 18:16:16

RMAN-07536: command not allowed when connected to a Pluggable Database

RMAN>

You get an error message because you must be connected to the CDB root to configure any recovery settings.

1. Exit RMAN.

RMAN> **EXIT**

1. Back up the USERS tablespace in ORCLPDB2.
   1. Connect to ORCLPDB2 as the SYS user.

$ **export** **ORACLE\_PDB\_SID=ORCLPDB2**

$ **rman target SYS/fenago**

…

connected to target database: ORCLCDB:ORCLPDB2 (DBID=1621666632)

RMAN>

* 1. Back up the USERS tablespace.

RMAN> **BACKUP TABLESPACE users;**

Starting backup at 2019-06-20:18:18:13

using target database control file instead of recovery catalog

allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=32 device type=DISK

channel ORA\_DISK\_1: starting full datafile backup set channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00016 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:18:18:14 channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:18:18:15

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B419BF707E 73E0536210ED0A54C7/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T 181814\_gjqmh6q3\_.bkp tag=TAG20190620T181814 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:01 Finished backup at 2019-06-20:18:18:15

RMAN>

* 1. Exit RMAN.

RMAN> **EXIT**

…

$

1. Can you connect to the CDB root and perform the same operation?
   1. Start RMAN and connect to the CDB root as the SYS user.

$ **export** **ORACLE\_PDB\_SID=**

$ **rman target /**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

* 1. Back up the USERS tablespace in ORCLPDB2. You must specify the PDB in which the tablespace exists.

RMAN> **BACKUP TABLESPACE ORCLPDB2:users;**

Starting backup at 2019-06-20:18:20:10

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=32 device type=DISK

channel ORA\_DISK\_1: starting full datafile backup set channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00016 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf

channel ORA\_DISK\_1: starting piece 1 at 2019-06-20:18:20:11 channel ORA\_DISK\_1: finished piece 1 at 2019-06-20:18:20:12

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B419BF707E 73E0536210ED0A54C7/backupset/2019\_06\_20/o1\_mf\_nnndf\_TAG20190620T 182010\_gjqmlv5s\_.bkp tag=TAG20190620T182010 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:01

Finished backup at 2019-06-20:18:20:12

Starting Control File and SPFILE Autobackup at 2019-06- 20:18:20:12

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011464412\_gjqmlwjy\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:18:20:13

RMAN>

**Question:** Did the operation back up only the tablespace data files?

**Answer:** No. It also backed up the SPFILE and control file. It is only when you are connected to the CDB root to perform backups that the SPFILE and control file are backed up.

* 1. Exit RMAN and close the terminal window.

RMAN> **EXIT**

$

**Practice 5-5: Configuring Block Change Tracking**

### Overview

In this practice, you will configure Block Change Tracking (BCT). Although BCT is optional, it reduces the time required for an incremental backup from the time to scan all the blocks in the database to a time proportional to the number of blocks that have changed since the last backup.

The BCT file can contain only 8 bitmaps, so the backup cannot be optimized if there have been more than 8 incremental backups since the parent level backup that the new incremental will be based on. Consider the 8-bitmap limit when developing your incremental backup strategy. For example, if you make a level 0 database backup followed by 7 differential incremental backups, the block change tracking file now includes 8 bitmaps. If you then make a cumulative level 1 incremental backup, RMAN cannot optimize the backup because the bitmap corresponding to the parent level 0 backup is overwritten with the bitmap that tracks the current changes.

### Assumptions

You have a terminal window open with variables set for the ORCLCDB database instance.

#### Time Estimate

It should take approximately 5 minutes to complete this practice.

### Tasks

1. Configure block change tracking to place the BCT file in the default data file creation destination.
   1. Start SQL\*Plus and connect to your ORCLCDB database with administrator privileges.

$ **sqlplus / as sysdba**

… SQL>

* 1. Set the DB\_CREATE\_FILE\_DEST initialization parameter to

/u01/app/oracle/oradata/ORCLCDB.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SQL> **ALTER SYSTEM SET DB\_CREATE\_FILE\_DEST =**  **'/u01/app/oracle/oradata/ORCLCDB';**  System altered.  SQL> **SHOW PARAMETER DB\_CREATE\_FILE\_DEST** | | | | |
| NAME |  | TYPE |  | VALUE |
| db\_create\_file\_dest  SQL> |  | string |  | /u01/app/oracle/oradata/ORCLCDB |

* 1. Enable block change tracking by using the ALTER DATABASE statement.

SQL> **ALTER DATABASE ENABLE BLOCK CHANGE TRACKING;**

Database altered. SQL>

**Note:** In another practice, you will disable block change tracking again, because it is not required for the practice environment.

1. Exit SQL\*Plus.

SQL> **exit**

…

$

**Practice 5-6: Using Incremental Backups**

### Overview

In this practice, you create an image copy of the target database for use as a base for an incremental backup strategy. You perform an incremental level 1 backup, and apply the incremental to the level 0 backup.

**Note:** Applying the incremental backup to the level 0 image copies makes the level 0 backup the same as if you had taken a level 0 instead of level 1.

### Assumptions

You have two terminal windows open. Environment variables in both windows are set for the

ORCLCDB database instance.

#### Time Estimate

It should take approximately 45 minutes to complete this practice.

### Tasks

1. Perform the following steps to check if the ORCLPDB1 pluggable database is open, and open it if it is not.
   1. Log in to SQL\*Plus as SYSDBA.

$ **sqlplus / as sysdba**

… SQL>

* 1. Check the open mode of the PDBs. If ORCLPDB1 is open in READ WRITE mode, exit from SQL\*Plus and proceed to step 2.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| SQL> **show pdbs** | | | | | | | | |
| CON\_ID |  |  | CON\_NAME |  | OPEN | MODE |  | RESTRICTED |
|  | 2 |  | PDB$SEED |  | READ | ONLY |  | NO |
|  | 3 |  | ORCLPDB1 |  | READ | WRITE |  | NO |
| SQL> | 4 |  | ORCLPDB2 |  | READ | WRITE |  | NO |

* 1. Optionally, if ORCLPDB1 is MOUNTED, open the PDB and then verify the change.

SQL> **alter pluggable database orclpdb1 open;**

Pluggable database altered.

**Note:** **You can ignore if you get already open ERROR**

SQL> **show pdbs**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| CON\_ID |  |  | CON\_NAME |  | OPEN | MODE |  | RESTRICTED |
|  | 2 |  | PDB$SEED |  | READ | ONLY |  | NO |
|  | 3 |  | ORCLPDB1 |  | READ | WRITE |  | NO |
|  | 4 |  | ORCLPDB2 |  | READ | WRITE |  | NO |

* 1. Exit from SQL\*Plus.

SQL> **exit**

…

$

1. Execute the DBMod\_Backup\_CrINVENTORY.sh script from the

/home/oracle/labs/DBMod\_Backup directory. This script creates the INVENTORY tablespace, the INVENTORY user, and populates the schema in NOLOGGING mode. (The execution of the script might take a while because several tables with over 90,000 rows are created. Your row count might be different.)

$ **cd /home/oracle/labs/DBMod\_Backup** [DBMod\_Backup]$ **./DBMod\_Backup\_CrINVENTORY.sh** SQL> Connected.

SQL> SQL>

Session altered.

SQL> SQL> DROP TABLESPACE INVENTORY INCLUDING CONTENTS and DATAFILES CASCADE CONSTRAINTS

\*

ERROR at line 1:

ORA-00959: tablespace 'INVENTORY' does not exist

…

SQL> 2 3 4 5 6 7 8 9 10

72982 rows created.

…

[DBMod\_Backup]$

**Note:** This script can be run multiple times, but note that it completely resets the

INVENTORY schema each time it runs.

1. Create a directory named /home/oracle/backup/orclcdb and then back up the ORCLCDB database to this directory. Use the RMAN command line and set the incremental level to 0.
   1. Create the new directory.

$ **mkdir -p /home/oracle/backup/orclcdb**

$

* 1. Perform a database incremental level 0 backup. Allocate a channel to use the

/home/oracle/backup/orclcdb directory for the data file copies destination.

#### $ rman target "'/ as sysbackup'"

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN> **run {**

#### ALLOCATE CHANNEL "ch1" DEVICE TYPE DISK FORMAT

#### "/home/oracle/backup/orclcdb/%U";

#### BACKUP AS COPY TAG 'BASE01' INCREMENTAL LEVEL 0 DATABASE;

**}**

using target database control file instead of recovery catalog allocated channel: ch1

channel ch1: SID=262 device type=DISK

Starting backup at 2019-06-20:21:03:40 channel ch1: starting datafile copy

…

Starting Control File and SPFILE Autobackup at 2019-06- 20:21:05:01

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011474301\_gjqx7xj6\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:21:05:02

released channel: ch1

RMAN>

As the backup is performed, notice where the backup files are created.

1. Open a new terminal window and start a workload that updates your database by executing the DBMod\_Backup\_workload.sh script.

$ **su - oracle**

$ **cd /home/oracle/labs/DBMod\_Backup**

$ **./DBMod\_Backup\_workload.sh**

… SQL>

Session altered.

SQL> Connected.

SQL> SQL> SQL> SQL> SQL>

2190 rows updated.

2

3

4

SQL>

Commit complete.

…

$

**Note:** The script updates over 2000 rows.

1. In your RMAN terminal session, perform an incremental level 1 backup to the

/home/oracle/backup/orclcdb directory.

RMAN> **run {**

**ALLOCATE CHANNEL "ch1" DEVICE TYPE DISK FORMAT**

**"/home/oracle/backup/orclcdb/%U";**

**BACKUP TAG 'incr\_update' INCREMENTAL LEVEL 1 DATABASE;**

**}**

…

allocated channel: ch1

channel ch1: SID=262 device type=DISK

Starting backup at 2019-06-20:21:10:34

channel ch1: starting incremental level 1 datafile backup set

…

Starting Control File and SPFILE Autobackup at 2019-06- 20:21:10:38

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011474638\_gjqxlgh4\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:21:10:39

released channel: ch1

RMAN>

**Note:** The time of the incremental data file backup is faster than the original level 0 and some data files do not need to be backed up at all.

1. Apply the incremental backups to the existing level 0 backup. Notice the recovery of the data file copy and the restore operation.

RMAN> **run {**

**ALLOCATE CHANNEL "ch1" DEVICE TYPE DISK FORMAT**

**"/home/oracle/backup/orclcdb/%U";**

**RECOVER COPY OF DATABASE WITH TAG 'BASE01';**

**}**

…

Starting recover at 2019-06-20:21:12:44 no copy of datafile 5 found to recover no copy of datafile 6 found to recover no copy of datafile 8 found to recover

channel ch1: starting incremental datafile backup set restore channel ch1: specifying datafile copies to recover

recovering datafile copy file number=00001 name=/home/oracle/backup/orclcdb/data\_D-ORCLCDB\_I-2778750799\_TS- SYSTEM\_FNO-1\_1ju4jnpc

…

Finished recover at 2019-06-20:21:12:48

Starting Control File and SPFILE Autobackup at 2019-06- 20:21:12:48

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011474768\_gjqxpjsb\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 20:21:12:49

released channel: ch1

RMAN>

1. Disable block change tracking. It does not need to be enabled for any other practices.

RMAN> **ALTER DATABASE DISABLE BLOCK CHANGE TRACKING;**

**Note: You will get error if it is already disabled. You can ignore the error**

1. Exit RMAN and remove your test data by executing the

DBMod\_Backup\_cleanupINVENTORY.sh script.

RMAN> **exit**

$ **cd /home/oracle/labs/DBMod\_Backup**

$ **./DBMod\_Backup\_cleanupINVENTORY.sh**

1. Optionally, use your other terminal window to view the output in the /tmp/cleanup.log

file while the DBMod\_Backup\_cleanupINVENTORY.sh script is executing.

$ **cat /tmp/cleanup.log**

1. Use the RMAN DELETE OBSOLETE command to delete obsolete backups. Be sure to respond YES when prompted.

$ **rman target "'/ as sysbackup'"**

…

connected to target database: ORCLCDB (DBID=2778750799

RMAN> **delete obsolete;**

…

Datafile Copy 45 2019-06-20:21:12:47

/home/oracle/backup/orclcdb/data\_D-ORCLCDB\_I-2778750799\_TS- INVENTORY\_FNO-17\_22u4jnrr

Do you really want to delete the above objects (enter YES or NO)? **YES**

deleted datafile copy

datafile copy file name=/home/oracle/backup/orclcdb/data\_D- ORCLCDB\_I-2778750799\_TS-INVENTORY\_FNO-17\_22u4jnrr RECID=45 STAMP=1011474767

Deleted 40 objects

RMAN>

**Note:** Your output may be different depending on the repetition of practices.

1. Optionally, crosscheck all data file copies of the ORCLCDB database.

RMAN> **crosscheck datafilecopy all;** released channel: ORA\_DISK\_1 allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=43 device type=DISK validation succeeded for datafile copy

datafile copy file name=/home/oracle/backup/orclcdb/data\_D- ORCLCDB\_I-2778750799\_TS-SYSTEM\_FNO-1\_1ju4jnpc RECID=37 STAMP=1011474765

…

validation succeeded for datafile copy

datafile copy file name=/home/oracle/backup/orclcdb/data\_D- ORCLCDB\_I-2778750799\_TS-USERS\_FNO-16\_1ou4jnqn RECID=47 STAMP=1011474767

Crosschecked 15 objects

RMAN>

1. Exit RMAN.

RMAN> **exit**

$

1. Close the second terminal window if you have not already done so.

**Practice 5-7: Backing Up Additional Database Files**

### Overview

In this practice, you create backups of important database files that are not part of the default backup set.

### Assumptions

You are using a terminal window with environment variables pointing to the ORCLCDB database instance.

#### Time Estimate

It should take approximately 15 minutes to complete this practice.

### Tasks

1. Back up the control file to a trace file. Then exit SQL\*Plus and view the trace file.

**Note:** This command creates a SQL script to re-create a control file.

* 1. Log in to SQL\*Plus as the SYSDBA user.

$ **sqlplus / as sysdba**

… SQL>

* 1. Use the ALTER DATABASE BACKUP CONTROLFILE TO TRACE SQL command.

SQL> **ALTER DATABASE BACKUP CONTROLFILE TO TRACE AS**

**'/home/oracle/backup/orclcdb/control.sql';**

Database altered. SQL>

* 1. Exit from SQL\*Plus.

SQL> **exit**

…

$

* 1. View the trace file. What can you learn from its content?

**Note:** The trace file requires some editing before it can be used.

$ **cat /home/oracle/backup/orclcdb/control.sql**

…

-- the database with the NORESETLOGS option and should be used only if

-- the current versions of all online logs are available. The second

-- set opens the database with the RESETLOGS option and should be used

-- if online logs are unavailable.

-- The appropriate set of statements can be copied from the trace into

-- a script file, edited as necessary, and executed when there is a

-- need to re-create the control file.

…

$

1. Back up the archive log files of the ORCLCDB database, deleting all the archive files after the backup is complete.
   1. Log in to RMAN as SYSBACKUP.

$ **rman target "'/ as sysbackup'"**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

* 1. Back up the archive log files and delete the files after the backup completes.

RMAN> **run {**

**allocate channel "CH1" DEVICE TYPE DISK FORMAT '/home/oracle/backup/orclcdb/%U';**

**backup archivelog all delete all input;**

**}**

…

Starting backup at 2019-06-20:21:34:57 current log archived

channel CH1: starting archived log backup set

…

channel CH1: backup set complete, elapsed time: 00:00:03 channel CH1: deleting archived log(s)

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_20/o1\_mf\_1\_40\_gjqz01js\_.arc RECID=16 STAMP=1011476099

Finished backup at 2019-06-20:21:35:02

…

RMAN>

* 1. Log out of RMAN.

RMAN> **exit**

…

$

**Practices for Lesson 6: Using**

**Optional Backup Features**

**Practices for Lesson 6: Overview**

### Overview

In this practice, you will use RMAN to create an archival backup.

**Practice 6-1: Creating an Archival Backup**

### Overview

In this practice, you create an archival backup, that is, a backup that is not under your normal retention policies and it is not in your normal fast recovery area (FRA) destination. This KEEP FOREVER database backup could be used for recovery if some of your later hands-on activities were to require it. Archival backups can be created with an open or a mounted database.

You must use a recovery catalog to be able to create an archival backup of an OPEN database. Without a recovery catalog, you must put the database in a MOUNTED state.

### Assumptions

You are using a terminal window as the oracle OS user, with environment variables pointing to the ORCLCDB database instance.

It is assumed that the database and listener are running. You can use the pgrep -lf smon command to verify that the database is started and the pgrep -lf tns command to verify that the listener is started. If you need to restart the database and listener, use the dbstart.sh script.

### Timing Estimate

It should take approximately 25 minutes to complete this practice.

### Tasks

1. Log in to RMAN.

$ **rman target "'/ as sysbackup'"**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. Shut down and then restart the database instance in MOUNT mode so you can create a backup copy of the entire database in a mounted state.

RMAN> **SHUTDOWN IMMEDIATE;**

using target database control file instead of recovery catalog database closed

database dismounted Oracle instance shut down

RMAN> **STARTUP MOUNT;**

connected to target database (not started)

Oracle instance started database mounted

…

RMAN>

1. Attempt to make an archival backup of your ORCLCDB database with the KEEP FOREVER

option.

RMAN> **BACKUP AS COPY DATABASE KEEP FOREVER;**

Starting backup at 2019-06-21:15:00:29

allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=260 device type=DISK RMAN-00571:

===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS

===============

RMAN-00571:

===========================================================

RMAN-03002: failure of backup command at 06/21/2019 15:00:30

RMAN-06522: KEEP FOREVER option is not supported without the recovery catalog

RMAN>

**Question:** Why did the backup command fail?

**Answer:** It failed because a backup with the KEEP FOREVER option requires the use of a recovery catalog. A recovery catalog is required when KEEP FOREVER is specified because backup records will eventually age out of the control file.

1. Attempt to make an archival backup of your ORCLCDB database that will be kept for 365 days.

RMAN> **BACKUP AS COPY DATABASE KEEP UNTIL TIME 'SYSDATE+365';**

Starting backup at 2019-06-21:15:14:05 using channel ORA\_DISK\_1

backup will be obsolete on date 2020-06-20:15:14:05

archived logs required to recover from this backup will be backed up

channel ORA\_DISK\_1: starting datafile copy input datafile file number=00001

name=/u01/app/oracle/oradata/ORCLCDB/system01.dbf

RMAN-03009: failure of backup command on ORA\_DISK\_1 channel at 06/21/2019 15:14:05

ORA-19811: cannot have files in DB\_RECOVERY\_FILE\_DEST with keep attributes

continuing other job steps, job failed will not be re-run

…

RMAN-00571:

===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS

===============

RMAN-00571:

===========================================================

RMAN-03002: failure of backup command at 06/21/2019 15:14:07

ORA-19811: cannot have files in DB\_RECOVERY\_FILE\_DEST with keep attributes

RMAN>

**Question:** Why did the backup command fail?

**Answer:** It failed because a backup with the KEEP option (an archival backup) cannot be written to the Fast Recovery Area. Allowing this has the potential of causing the Fast Recovery Area to quickly run out of space.

1. Create an archival backup with a FORMAT clause that will direct RMAN to create the backup in the /home/oracle/backup directory and keep it for 365 days.

#### RMAN> BACKUP DATABASE FORMAT '/home/oracle/backup/%U' TAG keep\_db\_tag KEEP UNTIL TIME 'SYSDATE+365';

Starting backup at 2019-06-21:15:39:34

using channel ORA\_DISK\_1

backup will be obsolete on date 2020-06-20:15:39:34

archived logs required to recover from this backup will be backed up

channel ORA\_DISK\_1: starting full datafile backup set channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00001 name=/u01/app/oracle/oradata/ORCLCDB/system01.dbf

…

using channel ORA\_DISK\_1

backup will be obsolete on date 2020-06-20:15:40:12 archived logs will not be kept or backed up

channel ORA\_DISK\_1: starting full datafile backup set channel ORA\_DISK\_1: specifying datafile(s) in backup set including current control file in backup set

channel ORA\_DISK\_1: starting piece 1 at 2019-06-21:15:40:13

channel ORA\_DISK\_1: finished piece 1 at 2019-06-21:15:40:14

piece handle=/home/oracle/backup/32u4lp6s\_1\_1 tag=KEEP\_DB\_TAG comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:01 Finished backup at 2019-06-21:15:40:14

RMAN>

1. Open the database.

RMAN> **alter database open;**

Statement processed RMAN>

1. Delete obsolete backups. Be sure to respond **YES** to the prompt.

RMAN> **delete obsolete;**

RMAN retention policy will be applied to the command RMAN retention policy is set to redundancy 1

using channel ORA\_DISK\_1

Deleting the following obsolete backups and copies:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Type |  | Key |  | Completion Time Filename/Handle |
| ---  Backup Set |  | 38 |  | 2019-06-20:21:12:48 |
| Backup Piece |  | 38 |  | 2019-06-20:21:12:48 |

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_06\_20

/o1\_mf\_s\_1011474768\_gjqxpjsb\_.bkp

|  |  |  |
| --- | --- | --- |
| Backup Set | 39 | 2019-06-20:21:24:02 |
| Backup Piece | 39 | 2019-06-20:21:24:02 |

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_06\_20

/o1\_mf\_s\_1011475441\_gjqycl1b\_.bkp

Do you really want to delete the above objects (enter YES or NO)? **YES**

deleted backup piece backup piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011474768\_gjqxpjsb\_.bkp RECID=38 STAMP=1011474768

deleted backup piece

backup piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_20/o1\_mf\_s\_1011475441\_gjqycl1b\_.bkp RECID=39 STAMP=1011475442

Deleted 2 objects

RMAN>

1. Exit RMAN.

RMAN> **exit**

…

$

**Practices for Lesson 7: Tuning RMAN Backup**

**Performance**

**Practices for Lesson 7: Overview**

### Overview

In these practices, you will monitor an RMAN backup job.

**Practice 7-1: Monitoring an RMAN Backup Job**

### Overview

In this practice, you will monitor an RMAN backup job by using the V$SESSION\_LONGOPS view.

### Assumptions

You are logged in as the oracle user.

It is assumed that the database and listener are running. You can use the pgrep -lf smon command to verify that the database is started and the pgrep -lf tns command to verify that the listener is started. If you need to restart the database and listener, use the dbstart.sh script.

#### Time Estimate

It should take approximately 10 minutes to complete this practice.

### Tasks

1. Log in to SQL\*Plus. See *Course Practice Environment: Security Credentials* for password information.

$ **sqlplus system/fenago**

… SQL>

1. Open a new terminal window and set the environment variables for the ORCLCDB database.

$ **. oraenv**

ORACLE\_SID = [orclcdb] ?

The Oracle base remains unchanged with value /u01/app/oracle

$

1. In the same terminal window, start Recovery Manager and connect to the CDB root (target database) as the SYS user.

$ **rman target /**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. Create a whole database backup.

RMAN> **BACKUP DATABASE;**

…

Starting backup at 2019-06-21:18:52:23

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=277 device type=DISK

channel ORA\_DISK\_1: starting full datafile backup set

…

Finished backup at 2019-06-21:18:53:01

Starting Control File and SPFILE Autobackup at 2019-06- 21:18:53:01

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_21/o1\_mf\_s\_1011552781\_gjt9wg1h\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 2019-06- 21:18:53:02

RMAN>

1. Return to your first terminal session and quickly execute the

$HOME/labs/DBMod\_Backup/DBMod\_Backup\_MonitorBackup.sql script approximately every 10-15 seconds to monitor the progress of the backup job.

SQL> **@$HOME/labs/DBMod\_Backup/DBMod\_Backup\_MonitorBackup**

SID

SERIAL#

CONTEXT

SOFAR TOTALWORK %\_COMPLETE

267

40731

5

0

638720

0

SQL> /

SID

SERIAL#

CONTEXT

SOFAR TOTALWORK %\_COMPLETE

267

40731

5

269440

638720

42.18

SQL> /

no rows selected SQL>

**Question:** How can you use this query to determine if there is something wrong with the backup job?

**Answer:** If you execute the query at intervals of 2 minutes or more and the %\_COMPLETE

column does not increase, then it is likely that RMAN is encountering a problem.

1. Return to your RMAN session. After the backup completes, log out of RMAN.

RMAN> **exit**

…

$

1. Return to your SQL\*Plus session. Log out of SQL\*Plus.

SQL> **exit**

…

$

1. Close all terminal windows opened for this practice.

**Practices for Lesson 8:**

**Recovery Catalog Overview**

**Practices for Lesson 8**

There are no practices for Lesson 8.

**Practices for Lesson 9:**

**Creating a Recovery Catalog**

## Practices for Lesson 9: Overview

### Overview

In these practices, you will perform the following one-time setup tasks:

* + Create a recovery catalog owner.
  + Create a recovery catalog.

## Practice 9-1: Creating a Recovery Catalog Owner

### Overview

In this practice, you create a user and grant appropriate privileges.

### Assumptions

You are in a terminal window with the environment variables pointing to the ORCLOEM database instance.

#### Timing Estimate

It should take approximately 20 minutes to complete this practice.

### Tasks

1. Open a terminal window. Use oraenv again to set the environment variables for the

ORCLOEM database.

$ **. oraenv**

ORACLE\_SID = [orclcdb] ? **orcloem**

The Oracle base remains unchanged with value /u01/app/oracle

$

1. Start the ORCLOEM database.
   1. Log in to SQL\*Plus as SYSDBA.

$ **sqlplus / as sysdba**

…

Connected to an idle instance.

SQL>

* 1. Start the instance.

SQL> **startup**

ORACLE instance started.

…

Database opened.

SQL>

1. Create a tablespace named RCATTBS for the recovery catalog.
   1. Use SQL\*Plus to log in to the ORCLOEM
   2. Create the RCATTBS tablespace with a single 15 MB datafile.

SQL> **CREATE TABLESPACE rcattbs DATAFILE**

**'/u01/app/oracle/oradata/ORCLOEM/rcat01.dbf' SIZE 15M REUSE;**

Tablespace created. SQL>

1. Create a user that will own the recovery catalog data. Name the user RCATOWNER. The default tablespace should be the RCATTBS tablespace, and the user should have unlimited quota on that tablespace. Refer to the *Course Practice Environment: Security Credentials* document for the correct password to replace *fenago.*

SQL> **CREATE USER rcatowner IDENTIFIED BY fenagoDEFAULT TABLESPACE rcattbs QUOTA unlimited on rcattbs;**

User created.

SQL>

1. Grant the RECOVERY\_CATALOG\_OWNER role to the RCATOWNER user and exit from SQL\*Plus.

SQL> **GRANT recovery\_catalog\_owner TO rcatowner;**

Grant succeeded.

SQL> **exit**

…

$

## Practice 9-2: Creating the Recovery Catalog

### Overview

In this practice, you use RMAN to create the recovery catalog in the RCATPDB pluggable database.

### Assumptions

A terminal window is open with the environment variables pointing to the ORCLOEM database instance.

#### Timing Estimate

It should take approximately 15 minutes to complete this practice.

### Tasks

1. Connect to the recovery catalog database by using RMAN. Log in as the recovery catalog owner you created in the previous practice. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

$ **rman catalog rcatowner**

…

recovery catalog database Password: **fenago**

connected to recovery catalog database

RMAN>

1. Create the recovery catalog. This command may take a few minutes to complete. When you see the RMAN> prompt, exit the RMAN session.

RMAN> **create catalog;**

recovery catalog created RMAN> **exit**

…

$

**Practices for Lesson 10: Managing Target Database Records**

## Practices for Lesson 10: Overview

### Overview

In these practices, you will perform the following one-time setup tasks:

* + Register the ORCLOEM database in the recovery catalog.

## Practice 10-1: Registering a Database in the Recovery Catalog

### Overview

In this practice, you use RMAN to register the ORCLOEM database in the recovery catalog that you created in a previous practice.

### Assumptions

A terminal window is open.

#### Timing Estimate

It should take approximately 15 minutes to complete this practice.

### Tasks

1. Set the environment variables for the ORCLOEM database.

$ **. oraenv**

ORACLE\_SID = [orcloem] ? **orcloem**

The Oracle base remains unchanged with value /u01/app/oracle

$

1. Use RMAN to connect to the target database (to be registered) and to the recovery catalog database. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

$ **rman target "'/ as sysbackup'" catalog rcatowner**

…

connected to target database: ORCLCDB (DBID=2778750799) recovery catalog database Password: **fenago**

connected to recovery catalog database

RMAN>

1. Register the database in the catalog.

RMAN> **register database;**

database registered in recovery catalog starting full resync of recovery catalog full resync complete

RMAN>

1. To verify that the registration was successful, execute the REPORT SCHEMA command. The output should list all of the data files associated with the ORCLOEM database that have been registered in the recovery catalog.

RMAN> **REPORT SCHEMA;**

Report of database schema for database with db\_unique\_name ORCLOEM

List of Permanent Datafiles

===========================

File Size(MB) Tablespace RB segs Datafile Name

1 920 SYSTEM YES

/u01/app/oracle/oradata/ORCLOEM/system01.dbf

1. 850 SYSAUX NO

...

List of Temporary Files

=======================

File Size(MB) Tablespace

…

1. Exit from RMAN.

RMAN> **exit**

…

$

## Practice 10-2: Configuring the Recovery Catalog for Recovery

### Overview

Your organization determined that if the recovery catalog is lost or damaged, it needs to be restored quickly and completely.

In this practice, you configure the retention policy for the recovery catalog (retaining two backups), configure the fast recovery area for ORCLOEM, enable archive log mode, and back up your ORCLOEM database.

### Assumptions

A terminal window is open.

#### Timing Estimate

It should take approximately 30 minutes to complete this practice.

### Tasks

1. Configure the retention policy with at least redundancy 2.
   1. Set your environment variables to point to the ORCLOEM database.

$ **. oraenv**

ORACLE\_SID = [orclcdb] ? **orcloem**

The Oracle base remains unchanged with value /u01/app/oracle

$

* 1. Start RMAN and use the recovery catalog database (ORCLOEM) as the target, with no catalog specified. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

**Note:** The recovery parameters are configured at the CDB level.

$ **rman target sys**

…

target database Password: **fenago**

connected to target database: ORCLOEM (DBID=2073894879)

RMAN>

* 1. Make sure that the retention policy for the recovery catalog is set to redundancy greater than 1. If it is not, set it to at least 2.

RMAN> **show retention policy;**

using target database control file instead of recovery catalog

RMAN configuration parameters for database with db\_unique\_name ORCLOEM are:

CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default

RMAN> **configure retention policy to redundancy 2;**

new RMAN configuration parameters:

CONFIGURE RETENTION POLICY TO REDUNDANCY 2;

new RMAN configuration parameters are successfully stored

RMAN>

* 1. Exit RMAN.

RMAN> **exit**

…

$

1. To configure the fast recovery area for ORCLOEM, perform the following:
   1. Ensure that the environment variables are set for the ORCLOEM database.

$ **. oraenv**

ORACLE\_SID = [orcloem] ? **orcloem**

The Oracle base remains unchanged with value /u01/app/oracle

$

* 1. Log in to SQL\*Plus.

$ **sqlplus / as sysdba**

… SQL>

* 1. Set the size of the fast recovery area for ORCLOEM to 12 GB.

SQL> **ALTER SYSTEM SET db\_recovery\_file\_dest\_size=12G SCOPE=BOTH;**

System altered.

SQL>

1. Verify that the recovery catalog database is in ARCHIVELOG mode.
   1. Check whether ORCLOEM is in ARCHIVELOG mode.

SQL> **archive log list**

Database log mode No Archive Mode

Automatic archival Disabled

Archive destination USE\_DB\_RECOVERY\_FILE\_DEST Oldest online log sequence 5

Current log sequence 7

SQL>

* 1. Shut down the database instance and restart it in MOUNT mode.

SQL> **SHUTDOWN IMMEDIATE**

Database closed. Database dismounted.

ORACLE instance shut down.

SQL> **STARTUP MOUNT**

…

Database mounted.

SQL>

* 1. Put the database in ARCHIVELOG mode and then open the database.

SQL> **ALTER DATABASE ARCHIVELOG;**

Database altered.

SQL> **ALTER DATABASE OPEN;**

Database altered. SQL>

* 1. Verify that the database is now in ARCHIVELOG mode.

SQL> **archive log list**

Database log mode Archive Mode

Automatic archival Enabled

Archive destination USE\_DB\_RECOVERY\_FILE\_DEST Oldest online log sequence 5

Next log sequence to archive 7 Current log sequence 7

SQL>

* 1. Exit from SQL\*Plus.

SQL> **exit**

...

$

1. Back up the recovery catalog database.
   1. Start RMAN and use the recovery catalog database as the target, with no catalog specified. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

$ **rman target sys**

…

target database Password: **fenago**

connected to target database: ORCLOEM (DBID=2073894879)

RMAN>

* 1. Back up the database. Review the output and find the data file for the tablespace that you created for the recovery catalog. **(Note: This command might take some time to complete)**

RMAN> **BACKUP DATABASE;**

starting backup at 2019-06-28:20:28:25

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=270 device type=DISK channel ORA\_DISK\_1: starting datafile copy

input datafile file number=00001 name=/u01/app/oracle/oradata/ORCLOEM/system01.dbf

output file name=/u01/app/oracle/fast\_recovery\_area/ORCLOEM/datafile/o1\_mf\_s

* 1. Delete all backups and Exit RMAN.

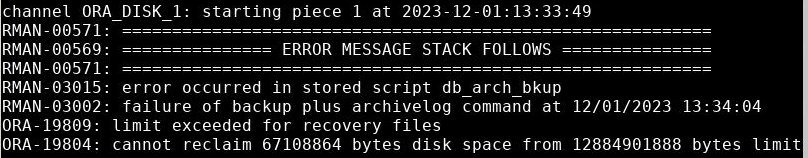
RMAN> **DELETE NOPROMPT BACKUP;**

RMAN> **exit**

…

$

**Note:** If you get limit following error in any of the exercises while creating backup, you can run the above command.



**Practices for Lesson 11:**

**Using Stored Scripts**

## Practices for Lesson 11: Overview

### Overview

In these practices, you will create and execute a stored script.

## Practice 11-1: Creating a Stored Script

### Overview

In this practice, you create a stored script.

### Assumptions

You have a terminal window open.

#### Timing Estimate

It should take approximately 10 minutes to complete this practice.

### Tasks

1. Set the environment variables for the target database, orcloem.

$ **. oraenv**

ORACLE\_SID = [ORCLOEM] ? **orcloem**

The Oracle base remains unchanged with value /u01/app/oracle

$

1. Invoke RMAN and connect to the target database and the ORCLOEM recovery catalog. Refer to the *Course Practice Environment: Security Credentials* document for the correct password for the recovery catalog owner.

$ **rman target / catalog rcatowner**

…

connected to target database: ORCLOEM (DBID=2778750799) recovery catalog database Password: \*\*\*\*\*\*

connected to recovery catalog database

RMAN>

1. Display the names of the stored scripts.

RMAN> **LIST SCRIPT NAMES;**

List of Stored Scripts in Recovery Catalog No scripts in recovery catalog

RMAN>

1. Create a local script named db\_arch\_bkup to perform a database backup, including the archived log files.

RMAN> **CREATE SCRIPT db\_arch\_bkup**

**{**

**BACKUP DATABASE PLUS ARCHIVELOG;**

**}**

created script db\_arch\_bkup RMAN>

1. List the contents of the script you created.

RMAN> **PRINT SCRIPT db\_arch\_bkup;**

printing stored script: db\_arch\_bkup

{

BACKUP DATABASE PLUS ARCHIVELOG;

}

RMAN>

1. Modify the db\_arch\_bu script so that it also deletes obsolete archived log files.

RMAN> **REPLACE SCRIPT db\_arch\_bkup**

**{**

**BACKUP DATABASE PLUS ARCHIVELOG;**

**DELETE OBSOLETE;**

**}**

replaced script db\_arch\_bkup RMAN>

1. Once again list the contents of the script.

RMAN> **PRINT SCRIPT db\_arch\_bkup;**

printing stored script: db\_arch\_bkup

{

BACKUP DATABASE PLUS ARCHIVELOG; DELETE OBSOLETE;

}

RMAN>

1. Log out of RMAN.

RMAN> **exit**

Recovery Manager complete.

$

## Practice 11-2: Executing a Stored Script

### Overview

In this practice, you execute a stored script.

### Assumptions

You have a terminal window. You successfully completed the practice to create a stored script named db\_arch\_bkup.

#### Timing Estimate

It should take approximately 15 minutes to complete this practice.

### Tasks

1. Set the environment variables for the target database, orcloem.

$ **. oraenv**

ORACLE\_SID = [orclcdb] ? **orcloem**

The Oracle base remains unchanged with value /u01/app/oracle

$

1. Invoke RMAN and connect to the target database and the ORCLOEM recovery catalog. Refer to the *Course Practice Environment: Security Credentials* document for the correct password for the recovery catalog owner.

$ **rman target / catalog rcatowner**

…

connected to target database: ORCLOEM (DBID=2778750799) recovery catalog database Password:

connected to recovery catalog database

RMAN>

1. Execute the db\_arch\_bkup script. Be sure to respond YES when prompted to delete the obsolete backups.

RMAN> **run**

**{**

**EXECUTE SCRIPT db\_arch\_bkup;**

**}**

executing script: db\_arch\_bkup

Starting backup at 2019-06-28:21:19:40 current log archived

allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=38 device type=DISK

channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set

input archived log thread=1 sequence=41 RECID=17 STAMP=1012157788

input archived log thread=1 sequence=42 RECID=18 STAMP=1012166381

channel ORA\_DISK\_1: starting piece 1 at 2019-06-28:21:19:43 channel ORA\_DISK\_1: finished piece 1 at 2019-06-28:21:19:46

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_28/o1\_mf\_annnn\_TAG20190628T211943\_gkf13hcp\_.bkp tag=TAG20190628T211943 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:03 Finished backup at 2019-06-28:21:19:46

…

Backup Piece 423 2019-06-28:21:19:45

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019\_06\_28/ o1\_mf\_annnn\_TAG20190628T211943\_gkf13hcp\_.bkp

Backup Set 436 2019-06-28:21:20:50

Backup Piece 442 2019-06-28:21:20:50

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857888C96117610E0536 210ED0A12EE/backupset/2019\_06\_28/o1\_mf\_nnndf\_TAG20190628T211946\_ gkf15cs1\_.bkp

Do you really want to delete the above objects (enter YES or NO)? **YES**

deleted archived log archived log file

name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_28/o1\_mf\_1\_41\_gkdrpsqt\_.arc RECID=17 STAMP=1012157788

…

backup piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857888C961176 10E0536210ED0A12EE/backupset/2019\_06\_28/o1\_mf\_nnndf\_TAG20190628T

211946\_gkf15cs1\_.bkp RECID=72 STAMP=1012166443

Deleted 13 objects

RMAN>

1. Verify that your script created a full database backup.

RMAN> **list backup;**

List of Backup Sets

===================

BS Key Time

Type LV Size

Device Type Elapsed Time Completion

299

Full

1.42G

DISK

00:00:03

2019-06-

21:15:39:37

BP Key: 319 KEEP\_DB\_TAG

Status: AVAILABLE

Compressed: NO Tag:

Piece Name: /home/oracle/backup/2tu4lp5m\_1\_1

Keep: BACKUP\_LOGS Until: 2020-06-20:15:39:34

…

List of Archived Logs in backup set 464

Thrd Seq Time

Low SCN

Low Time

Next SCN Next

1

43

3504011

2019-06-28:21:19:40 3504086

2019-

06-28:21:20:50

BS Key Time

Type LV Size

Device Type Elapsed Time Completion

476

Full

17.95M

DISK

00:00:01

2019-06-

28:21:20:54

BP Key: 478 TAG20190628T212053

Status: AVAILABLE

Compressed: NO Tag:

Piece Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_06\_28

/o1\_mf\_s\_1012166453\_gkf15oy7\_.bkp

SPFILE Included: Modification time: 2019-06-28:18:26:24 SPFILE db\_unique\_name: ORCLCDB

Control File Included: Ckp SCN: 3504115 Ckp time: 2019- 06-28:21:20:53

RMAN>

1. Exit RMAN.

RMAN> **exit**

Recovery Manager complete.

$

# Practices for Lesson 12: Creating and Using Virtual Private Catalogs

## Practices for Lesson 12: Overview

### Overview

In these practices, you will create a virtual catalog and use the virtual catalog when taking a backup.

## Practice 12-1: Enabling the Virtual Private Database (VPD) Functionality

### Overview

In this practice, you enable the Virtual Private Database (VPD) functionality.

RMAN uses the VPD functionality to implement virtual private catalogs. The VPD functionality is not enabled by default when the RMAN base recovery catalog is created. You need to explicitly enable the VPD model for a base recovery catalog by running the

$ORACLE\_HOME/rdbms/admin/dbmsrmanvpc.sql script.

### Assumptions

You have an open terminal window. You completed the practices to create the recovery catalog.

#### Timing Estimate

It should take approximately 20 minutes to complete this practice.

### Tasks

1. Set the environment variables for the recovery catalog database (ORCLOEM).

$ **. oraenv**

ORACLE\_SID = [orcldb] ? **orcloem**

The Oracle base remains unchanged with value /u01/app/oracle

$

1. Invoke SQL\*Plus and log in to RCATPDB as the SYSDBA user. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

$ s**qlplus / as sysdba**

…

…

SQL>

1. Execute the $ORACLE\_HOME/rdbms/admin/dbmsrmanvpc.sql script to enable the VPD model for all the virtual private catalogs of the base catalog schema, RCATOWNER.

SQL> **@$ORACLE\_HOME/rdbms/admin/dbmsrmanvpc.sql -vpd rcatowner**

Checking the operating user... Passed

Granting VPD privileges to the owner of the base catalog schema RCATOWNER

======================================== VPD SETUP STATUS:

VPD privileges granted successfully!

Connect to RMAN base catalog and perform UPGRADE CATALOG. Disconnected from Oracle Database …

$

1. Upgrade the recovery catalog.
   1. Invoke RMAN and connect to the recovery catalog database. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

$ **rman catalog rcatowner**

…

recovery catalog database Password: connected to recovery catalog database

RMAN>

* 1. Execute the UPGRADE CATALOG command. Be sure to enter the command a second time when prompted.

RMAN> **upgrade catalog;**

recovery catalog owner is RCATOWNER

enter UPGRADE CATALOG command again to confirm catalog upgrade RMAN> **upgrade catalog;**

recovery catalog upgraded to version 19.03.00.00.00 DBMS\_RCVMAN package upgraded to version 19.03.00.00 DBMS\_RCVCAT package upgraded to version 19.03.00.00.

RMAN>

* 1. Exit from RMAN.

RMAN> **exit**

…

$

**Practices for Lesson 13: Restore and Recovery Concepts**

## Practices for Lesson 13

There are no practices for Lesson 13.

**Practices for Lesson 14:**

**Diagnosing Failures**

## Practices for Lesson 14: Overview

### Overview

In these practices, you will use the Data Recovery Advisor to diagnose database failure.

## Practice 14-1: Diagnosing and Repairing Database Failure

### Overview

In this practice, you will use Data Recovery Advisor via the RMAN client interface to diagnose and repair the loss of a data file.

### Assumptions

You have two terminal windows open in which $HOME/labs/DBMod\_Recovery is the current directory, and environment variables point to the orclcdb database instance.

### Tasks

1. Open a terminal window and execute the rec\_config.sh script from the

$HOME/labs/DBMod\_Recovery/ directory. This script configures a minimal configuration to ensure that the database is recoverable. If you have already done the configuration the script does nothing and exits.

#### $ $HOME/labs/DBMod\_Recovery/rec\_config.sh

Verify controlfiles are multiplexed The control files are multiplexed Verify Pfile Exists

PFILE

/u01/app/oracle/product/19.3.0/dbhome\_1/dbs/initorclcdb.ora exists

Verify database is in Archivelog ARCHIVELOG

Database in Archivelog

Verify fast recovery area is configured

PL/SQL procedure successfully completed.

Verify if redo logfiles mulitplexed file\_name is

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo1a.log sqlstr is ALTER DATABASE ADD LOGFILE MEMBER

'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo1a.log ' TO GROUP 1

file\_name is

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo2a.log sqlstr is ALTER DATABASE ADD LOGFILE MEMBER

'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo2a.log ' TO GROUP 2

file\_name is

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo3a.log sqlstr is ALTER DATABASE ADD LOGFILE MEMBER

'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo3a.log ' TO GROUP 3

$

1. Execute the setup\_02\_01.sh script. This script creates the BAR91TBS tablespace in the file system, the BAR91 user, and the BARCOPY table, and populates the table. The script continues and takes a backup of the tablespace and updates the table. You can view the script output in the /tmp/setup.log file.

$ **cd $HOME/labs/DBMod\_Recovery**

$ **./setup\_02\_01.sh**

$

1. Execute the break\_02\_01.sh script. This script deletes a data file and causes the database to fail. You can view the script output in the /tmp/break.log file.

$ **./break\_02\_01.sh**

$

1. Open a new terminal window, as the oracle user with your environment variables pointing to orclcdb. Attempt to start the instance and open the ORCLPDB1 pluggable database. Observe the error messages, then exit SQL\*Plus.

**Note:** The error messages may occur while trying to open the ORCLCDB database or while trying to open the ORCLPDB1 PDB, or possibly both.

$ **sqlplus / as sysdba**

…

Connected to an idle instance.

SQL> **startup**

ORACLE instance started.

...

SQL> **show pdbs**

CON\_ID CON\_NAME

OPEN MODE RESTRICTED

1. PDB$SEED
2. ORCLPDB1
3. ORCLPDB2

READ ONLY NO MOUNTED

MOUNTED

SQL> **alter pluggable database orclpdb1 open;**

alter pluggable database orclpdb1 open

\*

ERROR at line 1:

ORA-01157: cannot identify/lock data file 190 - see DBWR trace file

ORA-01110: data file 190: '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bar91tbs01.dbf'

SQL> **exit**

$

1. Use the Data Recovery Advisor to list the database failure. **Note:** You might see one or more failure IDs.

$ **rman target "'/ as sysbackup'"**

…

RMAN> **LIST FAILURE;**

using target database control file instead of recovery catalog Database Role: PRIMARY

List of Database Failures

=========================

Failure ID Priority Status

Time Detected Summary

62

CRITICAL OPEN

02-JUL-19

System datafile 9:

'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf' needs media recovery

62 HIGH OPEN 02-JUL-19 One or more non- system datafiles need media recovery

62 HIGH OPEN 02-JUL-19 One or more non- system datafiles are missing

RMAN>

1. Use the Data Recovery Advisor to obtain advice on how to repair the failure. **Note:** During a manual recovery**,** if you see more than one optional manual action, you would focus on the one that references bar91tbs01.dbf, the missing file.

RMAN> **ADVISE FAILURE;**

Database Role: PRIMARY

List of Database Failures

=========================

Failure ID Priority Status

Time Detected Summary

62 CRITICAL OPEN 02-JUL-19 System datafile 9: '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf' needs media recovery

62 HIGH OPEN 02-JUL-19 One or more non- system datafiles need media recovery

62 HIGH OPEN 02-JUL-19 One or more non- system datafiles are missing

analyzing automatic repair options; this may take some time allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=40 device type=DISK analyzing automatic repair options complete

Mandatory Manual Actions

========================

no manual actions available

Optional Manual Actions

=======================

1. If you restored the wrong version of data file

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf, then replace it with the correct one

1. Automatic repairs may be available if you shutdown the database and restart it in mount mode
2. If you restored the wrong version of data file

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf, then replace it with the correct one

1. If you restored the wrong version of data file

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf, then replace it with the correct one

1. If you restored the wrong version of data file

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf, then replace it with the correct one

1. If file

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bar91tbs01.dbf was unintentionally renamed or moved, restore it

Automated Repair Options

========================

Option Repair Description

1 Restore and recover datafile 19 …

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_991361536.hm

RMAN>

1. Use the Data Recovery Advisor to repair the failure. Examine the contents of the repair script that was generated for this repair before it executes. Enter Y or YES when prompted to execute the script.

RMAN> **repair failure;**

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_991361536.hm

contents of repair script:

# restore and recover datafile

sql 'ORCLPDB1' 'alter database datafile 19 offline'; restore ( datafile 19 );

recover datafile 19;

sql 'ORCLPDB1' 'alter database datafile 19 online'; # recover datafile

sql 'ORCLPDB1' 'alter database datafile 9, 10, 11, 12 offline';

recover datafile 9, 10, 11, 12;

sql 'ORCLPDB1' 'alter database datafile 9, 10, 11, 12 online';

Do you really want to execute the above repair (enter YES or NO)? **YES**

executing repair script

...

channel ORA\_DISK\_1: restoring datafile 00190

input datafile copy RECID=19 STAMP=1012576702 file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8C28E6F854EB7DBB E0536210ED0AFDD9/datafile/o1\_mf\_bar91tbs\_gkpxfy1o\_.dbf

destination for restore of datafile 00190:

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bar91tbs01.dbf

...

repair failure complete

RMAN>

1. Open the ORCLPDB1 pluggable database.

RMAN> **alter pluggable database orclpdb1 open;**

Statement processed RMAN> **exit**

…

$

1. Confirm that ORCLPDB1 is open.
2. Return to the first terminal window. Clean up the environment following this practice by executing the cleanup\_02\_01.sh. script. You can view the script output in the

$ **sqlplus / as sysdba**

**…**

SQL> **show PDBS**

CON\_ID CON\_NAME

OPEN MODE RESTRICTED

1. PDB$SEED
2. ORCLPDB1
3. ORCLPDB2

READ ONLY NO READ WRITE NO

MOUNTED

SQL> **exit**

$

/tmp/cleanup.log file.

$ **./cleanup\_02\_01.sh**

$

1. Close the second terminal window.

**Practices for Lesson 15: Performing Complete Recovery**

## Practices for Lesson 15: Overview

### Overview

In these practices, you will initiate a recovery operation by using RMAN commands. You will use the Data Recovery Advisor to recover a datafile.

## Practice 15-1: Recovering from the Loss of a System-Critical Data File

### Overview

In this practice, you recover your CDB after the data file for the SYSTEM tablespace (in the CDB root) has been inadvertently removed.

#### Tip

Because you use several windows at the same time in this practice, you may find it helpful to change the name of each of them in their banner at the top.

To set a title for a terminal window:

1. In the terminal window's menu, select **Terminal** and then **Set Title**. A Set Title dialog box is displayed.
2. In the Title box, enter the window number.
3. Click **OK**.

### Assumptions

* + You are logged in as the oracle user.
  + You have configured the database for recovery.
  + It is assumed that the database and listener are running. You can use the pgrep -lf smon command to verify that the database is started and the pgrep -lf tns command to verify that the listener is started.

### Tasks

#### Configure the Database for Recovery

1. Execute the rec\_config.sh script from the $HOME/labs/DBMod\_Recovery/ directory. This script configures a minimal configuration to ensure that the database is recoverable. If you have already done the configuration the script does nothing and exits.

$ **$HOME/labs/DBMod\_Recovery/rec\_config.sh**

Verify controlfiles are multiplexed The control files are multiplexed Verify Pfile Exists

PFILE

/u01/app/oracle/product/19.3.0/dbhome\_1/dbs/initorclcdb.ora exists

Verify database is in Archivelog ARCHIVELOG

Database in Archivelog

Verify fast recovery area is configured

PL/SQL procedure successfully completed. Verify if redo logfiles mulitplexed

Redo log group 1 is Multiplexed Redo log group 2 is Multiplexed Redo log group 3 is Multiplexed

$

1. Perform a cold database backup.
   1. Create a directory to preserve the backup.

$ **mkdir -p /u01/app/oracle/donttouch**

* 1. Start RMAN.

$ **rman target /**

**...**

connected to target database: ORCLCDB (DBID=2778750799)

(Make sure you are not connected to PDB)

* 1. Shut down the database, and bring it to the mount stage.

RMAN> **SHUTDOWN IMMEDIATE;**

...

RMAN> **STARTUP MOUNT;**

...

* 1. Perform the backup to a separate location that will be not be accessed unless the practice fails.

RMAN> **BACKUP DATABASE FORMAT**

**"/u01/app/oracle/donttouch/backup\_%U";**

...

* 1. Open the database.

RMAN> **ALTER DATABASE OPEN;**

* 1. Exit RMAN.

RMAN> **EXIT**

1. Create a hot backup to the RMAN configured location.
   1. The default location for the automatic channel allocation is the

db\_recovery\_file\_dest.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| $ **sqlplus / as sysdba**  SQL> **show parameter db\_recovery\_file\_dest** | | | | |
| NAME |  | TYPE |  | VALUE |
| db\_recovery\_file\_dest |  | string |  | /u01/app/oracle/fast\_reco |

very\_area db\_recovery\_file\_dest\_size big integer 15G

SQL> **EXIT**

* 1. Perform the backup to the default location.

#### $ rman target /

...

connected to target database: ORCLCDB (DBID=2778750799)

RMAN> **BACKUP DATABASE;**

Starting backup at 27-JUN-19 allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=280 device type=DISK

channel ORA\_DISK\_1: starting full datafile backup set channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00003 name=/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf

input datafile file number=00001 name=/u01/app/oracle/oradata/ORCLCDB/system01.dbf

input datafile file number=00004 name=/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf

input datafile file number=00007 name=/u01/app/oracle/oradata/ORCLCDB/users01.dbf

channel ORA\_DISK\_1: starting piece 1 at 27-JUN-19

channel ORA\_DISK\_1: finished piece 1 at 27-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_27/o1\_mf\_nnndf\_TAG20190627T160931\_gk9tkztr\_.bkp tag=TAG20190627T160931 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:45 channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00161 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf

input datafile file number=00163 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

input datafile file number=00160 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf

input datafile file number=00162 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf

input datafile file number=00172 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/INVENTORY01.DBF

channel ORA\_DISK\_1: starting piece 1 at 27-JUN-19 channel ORA\_DISK\_1: finished piece 1 at 27-JUN-19

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8C28E6F854EB7D BBE0536210ED0AFDD9/backupset/2019\_06\_27/o1\_mf\_nnndf\_TAG20190627T 160931\_gk9tm8nb\_.bkp tag=TAG20190627T160931 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00014 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf

input datafile file number=00016 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf

input datafile file number=00013 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf

input datafile file number=00015 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf

channel ORA\_DISK\_1: starting piece 1 at 27-JUN-19 channel ORA\_DISK\_1: finished piece 1 at 27-JUN-19

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B419BF707E 73E0536210ED0A54C7/backupset/2019\_06\_27/o1\_mf\_nnndf\_TAG20190627T 160931\_gk9tmhpl\_.bkp tag=TAG20190627T160931 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00006 name=/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf

input datafile file number=00005 name=/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf

input datafile file number=00008 name=/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf

channel ORA\_DISK\_1: starting piece 1 at 27-JUN-19 channel ORA\_DISK\_1: finished piece 1 at 27-JUN-19

...

Starting Control File and SPFILE Autobackup at 27-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_27/o1\_mf\_s\_1012061437\_gk9tmych\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 27-JUN-19

* 1. Exit RMAN.

RMAN> **exit**

#### Create a Loss of a System-Critical Data File

#### Window 1

1. Use the same terminal window as *Window 1*.
2. Execute the $HOME/labs/DBMod\_Recovery/RMAN\_crash.sh shell script to remove the data file of the SYSTEM tablespace in the CDB root.

**Note:** This script may end with an error showing that data file system01.dbf cannot be opened.

$ **$HOME/labs/DBMod\_Recovery/RMAN\_crash.sh**

System altered.

System altered.

Select username from dba\_users where username = '1'

\*

ERROR at line 1:

ORA-00604: error occurred at recursive SQL level 1 ORA-01116: error in opening database file 1

ORA-01110: data file 1: '/u01/app/oracle/oradata/ORCLCDB/system01.dbf'

ORA-27041: unable to open file

Linux-x86\_64 Error: 2: No such file or directory Additional information: 3

$

1. Attempt an administrative task, such as creating a user.
   1. Start SQL\*Plus and connect to the CDB root as the SYSTEM user. Refer to *Course Practice Environment: Security Credentials* for the password value.

**Note:** At this point, any SQL statement gives an error.

$ **sqlplus system/fenago**

… ERROR:

ORA-00604: error occurred at recursive SQL level 1 ORA-01116: error in opening database file 1

ORA-01110: data file 1: '/u01/app/oracle/oradata/ORCLCDB/system01.dbf'

ORA-27041: unable to open file

Linux-x86\_64 Error: 2: No such file or directory Additional information: 3

ORA-00604: error occurred at recursive SQL level 2

ORA-01116: error in opening database file 1

ORA-01110: data file 1: '/u01/app/oracle/oradata/ORCLCDB/system01.dbf'

ORA-27041: unable to open file

Linux-x86\_64 Error: 2: No such file or directory Additional information: 3

Enter user-name: **^C**

$

1. Consider your recovery options.
   1. **Question:** Which type of recovery is possible in this case?

**Answer:** A complete recovery is possible as long as you have all available backups required. This means that you have a backup (backup set or image copy) of the missing data file, all archive log files required to recover the restored data file up to the current SCN of the CDB including all redo log files (one member in each group will be sufficient).

* 1. **Question:** Which methods can you use to recover the data file?
  2. **Answer:** RMAN is the best utility to recover data. You can use the RESTORE and

RECOVER commands or get help by using the LIST FAILURE command.

1. Determine the state of the database to choose the course of action in the next step. The output of step 3 matches with item b in this example.
   1. If the SQL\*Plus login displays "ORA-01034: ORACLE not available", then the database instance is down. ***Action:*** *Read case 3 in step 6.*
   2. If the SQL\*Plus login displays an error that includes "ORA-01116: error in opening database file 1", then database instance is still running, but impaired and must be terminated with shutdown abort. ***Action:*** *Read case 1 in step 6*
   3. If the rman target / command shows an error and immediately disconnects then the database instance is down.
   4. If RMAN connects but the restore datafile 1 command fails with errors about datafile 1, then the database is impaired and must be terminated with shutdown abort.
2. The exact commands that you will use vary based on the state of the database instance. Follow each case in order.
   1. Case 1: The database instance is not down, but impaired and must be terminated with shutdown abort.The database instance will likely shutdown automatically at some point. Use RMAN to shutdown abort the database instance, then skip to Case 3.

$ **rman target /**

RMAN> **Shutdown abort**

* 1. Case 2: The RMAN utility will not connect. Use SQL\*Plus to shutdown abort the database instance. Proceed to Case 3 to bring the database instance to the mount state using RMAN.

$ **sqlplus / as sysdba**

SQL> **SHUTDOWN ABORT**

SQL> **EXIT**

* 1. Case 3: The database instance is down. Use RMAN or SQL\*Plus to bring the database instance to the mount state.

$ **rman target /**

RMAN> **STARTUP MOUNT**

#### Recover the Database by Using the RESTORE and RECOVER Commands

#### Window 1

1. Restore the missing data file.

RMAN> **RESTORE DATAFILE 1;**

Starting restore at 27-JUN-19

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=256 device type=DISK

channel ORA\_DISK\_1: starting datafile backup set restore

channel ORA\_DISK\_1: specifying datafile(s) to restore from backup set

channel ORA\_DISK\_1: restoring datafile 00001 to

/u01/app/oracle/oradata/ORCLCDB/system01.dbf

channel ORA\_DISK\_1: reading from backup piece

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019\_06\_27/ o1\_mf\_nnndf\_TAG20190627T160931\_gk9tkztr\_.bkp

channel ORA\_DISK\_1: piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_27/o1\_mf\_nnndf\_TAG20190627T160931\_gk9tkztr\_.bkp tag=TAG20190627T160931

channel ORA\_DISK\_1: restored backup piece 1

channel ORA\_DISK\_1: restore complete, elapsed time: 00:00:07 Finished restore at 27-JUN-19

RMAN>

1. Recover the missing data file.

RMAN> **RECOVER DATAFILE 1;**

Starting recover at 27-JUN-19 using channel ORA\_DISK\_1

starting media recovery

media recovery complete, elapsed time: 00:00:00 Finished recover at 27-JUN-19

RMAN>

1. Open the CDB root.

RMAN> **ALTER DATABASE OPEN;**

Statement processed RMAN>

1. Open all PDBs.

RMAN> **ALTER PLUGGABLE DATABASE ALL OPEN;**

Statement processed RMAN>

1. Exit RMAN.

RMAN> **EXIT**

Recovery Manager complete.

$

1. Start SQL\*Plus and connect to the CDB root as the SYSDBA user.

$ **sqlplus / as sysdba**

… SQL>

1. Show pdb status.

SQL> **show pdbs**

CON\_ID CON\_NAME

OPEN MODE RESTRICTED

1. PDB$SEED
2. ORCLPDB1
3. ORCLPDB2

READ ONLY NO

READ WRITE NO

READ WRITE NO

1. Exit SQL\*Plus. Keep Window 1 open for the next section.

SQL> **Exit**

#### Use the Data Recovery Advisor to Recover the Database

1. **Window 2:** Open a new terminal window and execute the RMAN\_crash.sh script to create a failure. This window will be referred to as *Window 2*.

**Note:** This script may end with an error showing that data file system01.dbf cannot be opened.

$ **su - oracle**

$ **$HOME/labs/DBMod\_Recovery/RMAN\_crash.sh**

System altered.

System altered.

Select username from dba\_users where username = '1'

ERROR at line 1:

ORA-00604: error occurred at recursive SQL level 1 ORA-01116: error in opening database file 1

ORA-01110: data file 1: '/u01/app/oracle/oradata/ORCLCDB/system01.dbf'

ORA-27041: unable to open file

Linux-x86\_64 Error: 2: No such file or directory Additional information: 3

1. **Window 1:** Try to connect to the database.

$ **sqlplus / as sysdba**

…

1. If the database appears to be running attempt a simple query.

SQL> **select username from dba\_users;**

Note: You might get different based on database state. And this might work if you run this sql query immediately

select username from dba\_users

\*

ERROR at line 1:

ORA-00604: error occurred at recursive SQL level 1 ORA-01116: error in opening database file 1

ORA-01110: data file 1: '/u01/app/oracle/oradata/ORCLCDB/system01.dbf' ORA-27041: unable to open file

Linux-x86\_64 Error: 2: No such file or directory

Additional information: 3

1. **Window 1:** Shut down the database instance.

SQL> **shutdown abort**

1. **Window 1:** Exit SQL\*Plus.

SQL> **EXIT**

…

$

1. **Window 1:** Start RMAN and connect to the target database.

$ **rman target /**

…

connected to target database (not started)

RMAN>

1. **Window 1:** Start the database instance in MOUNT mode.

RMAN> **STARTUP MOUNT;**

database mounted

…

RMAN>

1. **Window 1:** Use the LIST FAILURE command to determine the error. The value in the

Summary column tells you that system01.dbf is missing.

RMAN> **LIST FAILURE;**

using target database control file instead of recovery catalog Database Role: PRIMARY

List of Database Failures

=========================

Failure ID Priority Status

Time Detected Summary

417

CRITICAL OPEN

27-JUN-19

System datafile 1:

'/u01/app/oracle/oradata/ORCLCDB/system01.dbf' is missing

RMAN>

1. **Window 1:** Display repair options. At the very end of the results, a repair script is listed.

RMAN> **ADVISE FAILURE;**

Database Role: PRIMARY List of Database Failures

=========================

Failure ID Priority Status

Time Detected Summary

417

CRITICAL OPEN

27-JUN-19

System datafile 1:

'/u01/app/oracle/oradata/ORCLCDB/system01.dbf' is missing

analyzing automatic repair options; this may take some time allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=19 device type=DISK analyzing automatic repair options complete

Mandatory Manual Actions

========================

no manual actions available

Optional Manual Actions

=======================

1. If file /u01/app/oracle/oradata/ORCLCDB/system01.dbf was unintentionally renamed or moved, restore it

Automated Repair Options

========================

Option Repair Description

1

Restore and recover datafile 1

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3469874817.hm

RMAN>

1. **Window 1:** Use the REPAIR FAILURE PREVIEW command to generate a script with all repair actions and comments.

RMAN> **REPAIR FAILURE PREVIEW;**

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3469874817.hm

contents of repair script:

# restore and recover datafile restore ( datafile 1 ); recover datafile 1;

sql 'alter database datafile 1 online';

RMAN>

1. **Window 1:** Use the REPAIR FAILURE command to repair database failures identified by the Data Recovery Advisor. When prompted, enter YES to execute the repair. When prompted to open the database, enter YES.

RMAN> **REPAIR FAILURE;**

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3469874817.hm

contents of repair script:

# restore and recover datafile restore ( datafile 1 ); recover datafile 1;

sql 'alter database datafile 1 online';

Do you really want to execute the above repair (enter YES or NO)? **YES**

executing repair script

Starting restore at 27-JUN-19 using channel ORA\_DISK\_1

channel ORA\_DISK\_1: starting datafile backup set restore

channel ORA\_DISK\_1: specifying datafile(s) to restore from backup set

channel ORA\_DISK\_1: restoring datafile 00001 to

/u01/app/oracle/oradata/ORCLCDB/system01.dbf

channel ORA\_DISK\_1: reading from backup piece

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019\_06\_27/ o1\_mf\_nnndf\_TAG20190627T160931\_gk9tkztr\_.bkp

channel ORA\_DISK\_1: piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_27/o1\_mf\_nnndf\_TAG20190627T160931\_gk9tkztr\_.bkp tag=TAG20190627T160931

channel ORA\_DISK\_1: restored backup piece 1

channel ORA\_DISK\_1: restore complete, elapsed time: 00:00:15 Finished restore at 27-JUN-19

Starting recover at 27-JUN-19 using channel ORA\_DISK\_1

starting media recovery

archived log for thread 1 with sequence 118 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_27

/o1\_mf\_1\_118\_gk9vttll\_.arc

archived log for thread 1 with sequence 119 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_27

/o1\_mf\_1\_119\_gk9vttmm\_.arc

archived log for thread 1 with sequence 120 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_27

/o1\_mf\_1\_120\_gk9wo0l1\_.arc

archived log for thread 1 with sequence 121 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_27

/o1\_mf\_1\_121\_gkb16db7\_.arc

archived log for thread 1 with sequence 122 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_27

/o1\_mf\_1\_122\_gkb16dck\_.arc

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_27/o1\_mf\_1\_118\_gk9vttll\_.arc thread=1 sequence=118

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_27/o1\_mf\_1\_119\_gk9vttmm\_.arc thread=1 sequence=119

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_27/o1\_mf\_1\_120\_gk9wo0l1\_.arc thread=1 sequence=120

media recovery complete, elapsed time: 00:00:00

Finished recover at 27-JUN-19

sql statement: alter database datafile 1 online repair failure complete

Do you want to open the database (enter YES or NO)? **YES**

database opened

RMAN>

1. **Window 1:** Open all the PDBs.

RMAN> **ALTER PLUGGABLE DATABASE ALL OPEN;**

Statement processed RMAN>

1. **Window 1:** Exit RMAN.

RMAN> **EXIT**

Recovery Manager complete.

$

1. **Window 2:** Close the second terminal window.

## Practice 15-2: Recovering from the Loss of an Application Data File

### Overview

In this practice, you recover a PDB data file that has been inadvertently removed.

### Assumptions

You are logged in as the oracle user and orclcdb and the listener are started.

### Tasks

#### Set Up Your Environment for the Practice

1. Execute the $HOME/labs/DBMod\_Recovery/setup\_pdb1.sh shell script. This script creates the TBS\_APP tablespace and ODR schema in ORCLPDB1. You can ignore object creation error messages.

$ **$HOME/labs/DBMod\_Recovery/setup\_pdb1.sh**

…

1 row created.

1 row created.

Commit complete.

$

1. Start RMAN and connect to the CDB root as the SYS user.

$ **rman target /**

…

connected to target database: ORCLCDB (DBID=1500451933)

RMAN>

1. Back up ORCLPDB1.

RMAN> **BACKUP PLUGGABLE DATABASE ORCLPDB1;**

Starting backup at 27-JUN-19

using target database control file instead of recovery catalog allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=35 device type=DISK

channel ORA\_DISK\_1: starting full datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00177 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/tbs\_app01.dbf

input datafile file number=00161 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf

input datafile file number=00163 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

input datafile file number=00160 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf

input datafile file number=00162 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf

input datafile file number=00172 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/INVENTORY01.DBF

channel ORA\_DISK\_1: starting piece 1 at 27-JUN-19

channel ORA\_DISK\_1: finished piece 1 at 27-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8C28E6F854EB7D BBE0536210ED0AFDD9/backupset/2019\_06\_27/o1\_mf\_nnndf\_TAG20190627T 182426\_gkb2gvt1\_.bkp tag=TAG20190627T182426 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 Finished backup at 27-JUN-19

Starting Control File and SPFILE Autobackup at 27-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_27/o1\_mf\_s\_1012069474\_gkb2h2gx\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 27-JUN-19

RMAN>

1. Exit RMAN.

RMAN> **EXIT**

Recovery Manager complete.

$

#### Remove a Data File

In this section, you run a script that removes a data file from ORCLPDB1. You research the problem and discover which data file is missing.

1. Execute the $HOME/labs/DBMod\_Recovery/RMAN\_crash\_app.sh script.

**Note:** A file that has been removed by the OS in Unix-based systems is still available to any application that has it open until the application closes the file. By default the Oracle database opens all the files when the database instance is opened, and holds them open.

A database health checker runs frequently, when then data file is detected as missing the PDB is aborted. An attempt to write to the file will also trigger detection.

$ **$HOME/labs/DBMod\_Recovery/RMAN\_crash\_app.sh**

System altered.

System altered.

$

1. Create an application table and insert data into it.
   1. Connect to the CDB root as the SYS user with the SYSDBA privilege.

$ **sqlplus / AS SYSDBA**

… SQL>

* 1. Connect to ORCLPDB1 as the SYS user.

SQL> **ALTER SESSION SET CONTAINER = ORCLPDB1;**

Session altered.

SQL>

* 1. Create a table named ODR.TEST. and insert a row into the table.

SQL> **CREATE TABLE odr.test (c NUMBER);**

Table created.

SQL> **INSERT into ODR.TEST VALUES ( 14 );**

insert into ODR.test values (14)

\*

ERROR at line 1:

ORA-01116: error in opening database file 177 ORA-01110: data file 177:

'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/tbs\_app01.dbf' ORA-27041: unable to open file

Linux-x86\_64 Error: 2: No such file or directory Additional information: 3

* 1. What is the status of the PDB?

SQL> **SHOW PDBs**

ERROR:

ORA-03135: connection lost contact Process ID: 21970

Session ID: 48 Serial number: 65393

SP2-1545: This feature requires Database availability.

**Note:** The output could also be as below, if the checker has not aborted the PDB yet.

* 1. Exit and reconnect to the database as sys.

SQL> **Exit**

…

$ **sqlplus / as sysdba**

…

SQL> **SHOW PDBS**

CON\_ID CON\_NAME

OPEN MODE RESTRICTED

1. PDB$SEED
2. ORCLPDB1
3. ORCLPDB2

READ ONLY NO

READ WRITE NO

READ WRITE NO

SQL> **SHOW PDBs**

CON\_ID CON\_NAME

OPEN MODE RESTRICTED

3 ORCLPDB1

READ WRITE NO

* 1. If PDB ORCLPDB is open, close it. This may show an error.

SQL> **ALTER PLUGGABLE DATABASE ORCLPDB1 CLOSE;**

Pluggable database altered.

* 1. Attempt to open the PDB.

SQL> **ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN;**

ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN

\*

ERROR at line 1:

ORA-01157: cannot identify/lock data file 177 - see DBWR trace file

ORA-01110: data file 177:

'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/tbs\_app01.dbf'

1. Exit SQL\*Plus.

SQL> **EXIT**

…

$

#### Restore and Recover ORCLPDB1

In this section, you use Recovery Manager to restore and recover ORCLPDB1.

1. **Question:** Which type of recovery is possible in this case?

**Answer:** A complete recovery is possible as long as you have all available backups required. This means that you have a backup (backup set or image copy) of the missing data file and all archive log files required to recover the restored data file up to the current SCN of the PDB including all redo log files (one member in each group will be sufficient).

1. **Question:** Which methods can you use to recover?

**Answer:** RMAN is the best utility to recover data. You can use the RESTORE and RECOVER commands, or you can get help with the LIST FAILURE commands. You can also use the simple REPAIR command.

#### Use the REPAIR Command

Let’s use RMAN's REPAIR command to perform all the necessary operations (for example, restore and recovery) to fully recover the data file.

1. Start RMAN and log in to the target database as the SYS user.

$ **rman target /**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. Execute the REPAIR command. This command restores and recovers the data file.

RMAN> **REPAIR PLUGGABLE DATABASE ORCLPDB1;**

Starting restore at 27-JUN-19

using target database control file instead of recovery catalog

allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=274 device type=DISK Executing: alter database datafile 160 offline Executing: alter database datafile 161 offline Executing: alter database datafile 162 offline Executing: alter database datafile 163 offline Executing: alter database datafile 172 offline Executing: alter database datafile 177 offline

channel ORA\_DISK\_1: starting datafile backup set restore

channel ORA\_DISK\_1: specifying datafile(s) to restore from backup set

channel ORA\_DISK\_1: restoring datafile 00160 to

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf

…

channel ORA\_DISK\_1: restoring datafile 00177 to

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/tbs\_app01.dbf

channel ORA\_DISK\_1: reading from backup piece

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8C28E6F854EB7DBBE0536 210ED0AFDD9/backupset/2019\_06\_27/o1\_mf\_nnndf\_TAG20190627T182426\_ gkb2gvt1\_.bkp

channel ORA\_DISK\_1: piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8C28E6F854EB7D BBE0536210ED0AFDD9/backupset/2019\_06\_27/o1\_mf\_nnndf\_TAG20190627T 182426\_gkb2gvt1\_.bkp tag=TAG20190627T182426

channel ORA\_DISK\_1: restored backup piece 1

channel ORA\_DISK\_1: restore complete, elapsed time: 00:00:15 Finished restore at 27-JUN-19

Starting recover at 27-JUN-19 using channel ORA\_DISK\_1

starting media recovery

archived log for thread 1 with sequence 124 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_27

/o1\_mf\_1\_124\_gkb2l6wf\_.arc

…

archived log for thread 1 with sequence 129 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_06\_27

/o1\_mf\_1\_129\_gkb9grvl\_.arc

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_27/o1\_mf\_1\_124\_gkb2l6wf\_.arc thread=1 sequence=124

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_27/o1\_mf\_1\_125\_gkb2l6xk\_.arc thread=1 sequence=125

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_27/o1\_mf\_1\_126\_gkb2lqsg\_.arc thread=1 sequence=126

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_ 06\_27/o1\_mf\_1\_127\_gkb2lqtd\_.arc thread=1 sequence=127

media recovery complete, elapsed time: 00:00:01 Executing: alter database datafile 160 online Executing: alter database datafile 161 online Executing: alter database datafile 162 online Executing: alter database datafile 163 online Executing: alter database datafile 172 online Executing: alter database datafile 177 online Finished recover at 27-JUN-19

RMAN>

1. Open ORCLPDB1.

RMAN> **ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN;**

Statement processed RMAN>

1. Exit RMAN.

RMAN> **EXIT**

Recovery Manager complete.

$

1. Try again to create a table named ODR.TEST in ORCLPDB1.
   1. Connect to the CDB root as the SYS user with the SYSDBA privilege.

$ **sqlplus / AS SYSDBA**

… SQL>

* 1. Connect to ORCLPDB1 as the SYS user.

SQL> **ALTER SESSION SET CONTAINER = ORCLPDB1;**

Session altered.

SQL>

* 1. Try to re-create the odr.test table.

SQL> **CREATE TABLE odr.test (c NUMBER);**

CREATE TABLE odr.test (c NUMBER)

\*

ERROR at line 1:

ORA-00955: name is already used by an existing object

SQL>

* 1. The table exists. Are there any rows?

SQL> **SELECT \* FROM ODR.TEST;**

no rows selected

**Note:** The table was not created because the table definition exists in the SYSTEM tablespace, but there are no rows because the tablespace TBS\_APP was missing when you attempted to insert the row.

1. Try to insert data into the ODR.TEST table again.
   1. Issue the INSERT command. The operation succeeds, which means the REPAIR

command recovered the data file.

SQL> **INSERT INTO odr.test VALUES (2);**

1 row created.

SQL>

* 1. Commit the transaction.

SQL> **commit;**

Commit complete.

SQL>

* 1. Exit SQL\*Plus.

SQL> **EXIT**

…

$

**Practices for Lesson 16:**

**Performing Point-in-Time Recovery**

## Practices for Lesson 16: Overview

### Overview

In this practice, you will recover a table from a backup set without affecting other objects in the tablespace or schema.

## Practice 16-1: Recovering a Table from a Backup

SQL> **SELECT NAME, LOG\_MODE, OPEN\_MODE FROM V$DATABASE;**

NAME

LOG\_MODE

OPEN\_MODE

ORCLCDB

ARCHIVELOG READ WRITE

### Overview

In this practice, you will recover a table from a backup set (without affecting other objects in the tablespace or schema). The tasks include the following:

* Set up your test environment and confirm the configuration, which typically is a one- time task.
* In RMAN, perform a level 0 backup plus archive logs and delete obsolete backups.
* In SQL\*Plus, create and populate a new TEST\_TABLE. Note the SCN after commit.
* In RMAN, perform a level 1 backup.
* In SQL\*Plus, create the need to recover a table by purging TEST\_TABLE.
* In RMAN, recover your test table to the SCN.
* In SQL\*Plus, confirm the success of the recovery.
* Clean up your practice environment.

### Assumptions

You have two terminal windows open in which you are logged in as the oracle OS user,

$HOME/labs/DBMod\_Recovery is the current directory, and environment variables point to the orclcdb database instance.

### Tasks

1. Prepare for this practice by executing the setup\_04\_02.sh script from the

$HOME/labs/DBMod\_Recovery directory. This script should be run as **oracle** user:

* + Creates a new tablespace and user
  + As the new user, creates a table and populates it
  + Saves its output in the /tmp/setup.log file.

$ **cd $HOME/labs/DBMod\_Recovery**

$ **./setup\_04\_02.sh**

1. Start a SQL\*Plus session and verify your test configuration.
2. Log in as the SYS user.

$ **sqlplus / as sysdba**

… SQL>

1. Confirm that the database is in ARCHIVELOG mode.

SQL>

1. Confirm that compatibility is set to 19.0 or higher.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SQL> **show parameter** | **compatible** |  | | | |
| NAME |  |  | TYPE |  | VALUE |
| compatible noncdb\_compatible  SQL> |  |  | string boolean |  | 19.0.0  FALSE |

1. Confirm your backup location and size.

SQL> **show parameter recovery\_file**

NAME

TYPE

VALUE

db\_recovery\_file\_dest

string

/u01/app/oracle/fast\_recovery\_area db\_recovery\_file\_dest\_size big integer 15G remote\_recovery\_file\_dest string

SQL>

1. Connect to ORCLPDB1, then confirm the setup by executing the lab\_04\_02a.sql script from the $HOME/labs/DBMod\_Recovery directory. The BAR user should own the BARCOPY table.

SQL> **alter session set container=ORCLPDB1;**

Session altered.

SQL> **@lab\_04\_02a.sql**

SQL> COL TABLE\_NAME

FORMAT A30

SQL> COL TABLESPACE\_NAME FORMAT A15 SQL> COL OWNER FORMAT A10 SQL>

SQL> SELECT TABLE\_NAME, TABLESPACE\_NAME, STATUS

1. FROM DBA\_TABLES
2. WHERE OWNER = 'BAR' 4 /

TABLE\_NAME

TABLESPACE\_NAME STATUS

BARCOPY

BARTBS

VALID

SQL>

1. In your second terminal window, start an RMAN session and connect to your orclcdb

database as the target instance.

**Note:** Some of the following steps generate a lot of output. The easiest way to send RMAN output both to a log file and to standard output is to use the Linux tee command or its equivalent. There is no need to do this if your standard output allows you to scroll as much as you wish. Using the following command example, you can view the output in the

/home/oracle/rman\_04.log file.

$ **rman target "'/ as sysbackup'"| tee /home/oracle/rman\_04.log**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN>

1. Confirm or configure autobackup of the control file and perform a level 0 backup. It will take few minutes to complete.

#### RMAN> show CONTROLFILE AUTOBACKUP;

using target database control file instead of recovery catalog

RMAN configuration parameters for database with db\_unique\_name ORCLCDB are:

CONFIGURE CONTROLFILE AUTOBACKUP ON; # default

#### RMAN> backup incremental level 0 database plus archivelog;

Starting backup at 29-JUN-19 current log archived allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=279 device type=DISK channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set

input archived log thread=1 sequence=140 RECID=33 STAMP=1012139222

input archived log thread=1 sequence=141 RECID=34 STAMP=1012139224

input archived log thread=1 sequence=142 RECID=35 STAMP=1012139226

…

input archived log thread=1 sequence=159 RECID=61 STAMP=1012243668

input archived log thread=1 sequence=160 RECID=59 STAMP=1012243664

channel ORA\_DISK\_1: starting piece 1 at 29-JUN-19

channel ORA\_DISK\_1: finished piece 1 at 29-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_29/o1\_mf\_annnn\_TAG20190629T203255\_gkhlqqtq\_.bkp tag=TAG20190629T203255 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:03 channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set

input archived log thread=1 sequence=118 RECID=12 STAMP=1012062683

…

input archived log thread=1 sequence=135 RECID=28 STAMP=1012138240

channel ORA\_DISK\_1: starting piece 1 at 29-JUN-19

channel ORA\_DISK\_1: finished piece 1 at 29-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_29/o1\_mf\_annnn\_TAG20190629T203255\_gkhlqtz1\_.bkp tag=TAG20190629T203255 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:07 channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set input archived log thread=1 sequence=1 RECID=62 STAMP=1012247798 channel ORA\_DISK\_1: starting piece 1 at 29-JUN-19

channel ORA\_DISK\_1: finished piece 1 at 29-JUN-19

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_29/o1\_mf\_annnn\_TAG20190629T203255\_gkhlr231\_.bkp tag=TAG20190629T203255 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:03 channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set input archived log thread=1 sequence=2 RECID=63 STAMP=1012249687

…

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:01 Finished backup at 29-JUN-19

Starting backup at 29-JUN-19 using channel ORA\_DISK\_1

channel ORA\_DISK\_1: starting incremental level 0 datafile backup set

channel ORA\_DISK\_1: specifying datafile(s) in backup set

input datafile file number=00003 name=/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf

channel ORA\_DISK\_1: finished piece 1 at 29-JUN-19

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_29/o1\_mf\_nnnd0\_TAG20190629T203311\_gkhlr7qg\_.bkp tag=TAG20190629T203311 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:25

…

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:15 Finished backup at 29-JUN-19

…

Starting Control File and SPFILE Autobackup at 29-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_29/o1\_mf\_s\_1012250063\_gkhlthwx\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 29-JUN-19 RMAN>

1. In your SQL\*Plus session, create and populate a new table named BAR.TEST\_TABLE by executing the lab\_04\_02b.sql script. Note the SCN after the commit.

SQL> **@lab\_04\_02b.sql**

… SQL>

SQL> -- \*\*\*\* Note the SCN!

SQL> SELECT NAME, CURRENT\_SCN FROM V$DATABASE;

NAME

CURRENT\_SCN

ORCLCDB

10631091

SQL> **SELECT \* FROM BAR.TEST\_TABLE**;

NUM NAME

NOW

1. First test row 03-JUL-19
2. Second test row 03-JUL-19
3. Third test row

#### Be sure to note the SCN value displayed in the CURRENT\_SCN column. You will use it for recovery!

1. In your RMAN session, perform a level 1 backup. If you started your RMAN session with the tee command, then your output is redirected to the /home/oracle/rman\_04.log file.

#### RMAN> backup incremental level 1 database plus archivelog;

Starting backup at 29-JUN-19 current log archived

using channel ORA\_DISK\_1

channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set

input archived log thread=1 sequence=140 RECID=33 STAMP=1012139222

…

Starting backup at 29-JUN-19 current log archived

using channel ORA\_DISK\_1

channel ORA\_DISK\_1: starting archived log backup set

channel ORA\_DISK\_1: specifying archived log(s) in backup set input archived log thread=1 sequence=7 RECID=68 STAMP=1012250592 channel ORA\_DISK\_1: starting piece 1 at 29-JUN-19

channel ORA\_DISK\_1: finished piece 1 at 29-JUN-19

piece handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019

\_06\_29/o1\_mf\_annnn\_TAG20190629T204312\_gkhmc0cl\_.bkp tag=TAG20190629T204312 comment=NONE

channel ORA\_DISK\_1: backup set complete, elapsed time: 00:00:01 Finished backup at 29-JUN-19

Starting Control File and SPFILE Autobackup at 29-JUN-19 piece

handle=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/201 9\_06\_29/o1\_mf\_s\_1012250593\_gkhmc1pv\_.bkp comment=NONE

Finished Control File and SPFILE Autobackup at 29-JUN-19

RMAN>

1. In your SQL\*Plus session, create the need to recover a table by purging it. Optionally, view your SCN before and after the DROP TABLE command.

SQL> **SELECT NAME, CURRENT\_SCN FROM V$DATABASE;**

NAME

CURRENT\_SCN

ORCLCDB

10631264

SQL> **drop table BAR.test\_table purge;**

1. Optionally, view the current tables that the BAR user owns. The TEST\_TABLE should not be displayed.

SQL> **SELECT table\_name FROM dba\_tables WHERE owner = 'BAR';**

TABLE\_NAME

BARCOPY SQL>

Table dropped.

SQL> **SELECT NAME, CURRENT\_SCN FROM V$DATABASE;**

NAME

CURRENT\_SCN

ORCLCDB

10631475

SQL>

1. In your RMAN session, recover your test table to **YOUR SCN** that you recorded in **Step 5**. Provide the following input with the RECOVER command:
   * Names of tables or table partitions to be recovered
   * SCN (or point in time) to which the tables or table partitions need to be recovered
   * Whether the recovered tables or table partitions must be imported into the target database (default is Yes.)
   * Auxiliary destination '/u01/app/oracle/backup/test'.

First, confirm that the directory of the auxiliary destination is empty and then execute your

RECOVER command. The recover command takes approximately 4 minutes.

RMAN> **HOST "ls /u01/app/oracle/backup/test/\*";**

ls: cannot access /u01/backup/test/\*: No such file or directory host command complete

RMAN-00571:

===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS

===============

RMAN-00571:

===========================================================

RMAN-06135: error executing host command: Additional information: 512

RMAN>

**Note:** This positive error prior to the RECOVER command confirms that the auxiliary destination is empty.

RMAN> **RECOVER TABLE BAR.TEST\_TABLE OF PLUGGABLE DATABASE ORCLPDB1 UNTIL SCN UPDATE\_HERE** <<Your SCN from Step 5

#### AUXILIARY DESTINATION '/u01/app/oracle/backup/test';

Starting recover at 29-JUN-19 using channel ORA\_DISK\_1

RMAN-05026: warning: presuming following set of tablespaces applies to specified point-in-time

List of tablespaces expected to have UNDO segments Tablespace SYSTEM

Tablespace ORCLPDB1:SYSTEM Tablespace UNDOTBS1 Tablespace ORCLPDB1:UNDOTBS1

Creating automatic instance, with SID='mcFq'

initialization parameters used for automatic instance: db\_name=ORCLCDB db\_unique\_name=mcFq\_pitr\_ORCLPDB1\_ORCLCDB

…

auxiliary instance file

/u01/app/oracle/backup/test/ORCLCDB/8C28E6F854EB7DBBE0536210ED0A FDD9/datafile/o1\_mf\_undotbs1\_gkhmrq50\_.dbf deleted

auxiliary instance file

/u01/app/oracle/backup/test/ORCLCDB/datafile/o1\_mf\_undotbs1\_gkhm qn2h\_.dbf deleted

auxiliary instance file

/u01/app/oracle/backup/test/ORCLCDB/8C28E6F854EB7DBBE0536210ED0A FDD9/datafile/o1\_mf\_system\_gkhmrq4t\_.dbf deleted

auxiliary instance file

/u01/app/oracle/backup/test/ORCLCDB/datafile/o1\_mf\_system\_gkhmqn 21\_.dbf deleted

auxiliary instance file

/u01/app/oracle/backup/test/ORCLCDB/controlfile/o1\_mf\_gkhmqfrp\_. ctl deleted

auxiliary instance file tspitr\_mcFq\_44423.dmp deleted Finished recover at 29-JUN-19

RMAN>

**Note:** RMAN uses your input to automate the process of recovering the specified table. RMAN performs the following tasks:

1. Determines the backup based on the SCN you provide
2. Creates an auxiliary instance
3. Recovers your tables or table partitions, up to the specified point in time, into this auxiliary instance
4. Creates a Data Pump export dump file that contains the recovered objects
5. Imports the recovered objects into the target database
6. Removes the auxiliary instance
7. Delete obsolete archive logs and then exit RMAN.

RMAN> **delete noprompt obsolete;**

RMAN retention policy will be applied to the command RMAN retention policy is set to redundancy 1

using channel ORA\_DISK\_1

Deleting the following obsolete backups and copies:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8857B419BF707E73E0536 210ED0A54C7/backupset/2019\_06\_27/o1\_mf\_nnndf\_TAG20190627T160931\_ gk9tmhpl\_.bkp

…

Deleted 9 objects

RMAN> **exit**

…

$

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Type |  | Key |  | Completion Time |  | Filename/Handle |
| Backup Set |  | 6 |  | 27-JUN-19 |  |  |
| Backup Piece |  | 6 |  | 27-JUN-19 |  |  |
| /u01/app/oracle/fast\_recovery\_area/ORCLCDB/backupset/2019\_06\_27/  o1\_mf\_nnndf\_TAG20190627T160931\_gk9tkztr\_.bkp  Backup Set 8 27-JUN-19 | | | | | | |
| Backup Piece | 8 | | 27-JUN-19 | | | |

1. In your SQL\*Plus session, query all rows of the test table to confirm the success of the recovery. Then exit SQL\*Plus.

SQL> **SELECT \* FROM BAR.TEST\_TABLE;**

NUM NAME

NOW

1. First test row 29-JUN-19
2. Second test row 29-JUN-19
3. Third test row

SQL> **exit**

…

$

1. Clean up the practice environment by executing the cleanup\_04\_02.sh script. This script removes the original and the transported tablespace, as well as the backup and dump files. The script saves its output in the /tmp/cleanup.log file.

$ **./cleanup\_04\_02.sh**

$

1. Keep all terminal windows open for the next practice.

**Practices for Lesson 17:**

**Performing Block Media Recovery**

## Practices for Lesson 17: Overview

### Overview

In these practices, you will use the Data Recovery Advisor to diagnose database failure.

## Practice 17-1: Repairing Block Corruption

### Overview

In this practice, you will use the Data Recovery Advisor to set up, discover, and repair a corrupted block in a data file.

### Assumptions

You have a terminal window open in which $HOME/labs/DBMod\_Recovery is the current directory, and environment variables point to the orclcdb database instance.

### Tasks

1. Prepare for this practice by executing the setup\_05\_01.sh script. This script creates the BC user, the BCTBS tablespace, and the BCCOPY table. To prepare for this practice the script populates the table, creates a backup, and updates the table. You can view the script output in the /tmp/setup.log file.

$ **cd $HOME/labs/DBMod\_Recovery**

$ **./setup\_05\_01.sh**

1. Corrupt a data file created in the previous step by executing the break\_05\_01.sql

script. When prompted, enter the block number that is displayed in the BLOCK\_NO column.

**Note:** The corrupt block error is expected. The script performs a query against the BCCOPY

table to force the corrupt block to be discovered.

$ **sqlplus /nolog @break\_05\_01.sql**

… Connected.

FILE\_NO BLOCK\_NO

18

129

System altered.

'Enter Block number when prompted' Enter value for block\_no: **129**

0+1 records in

0+1 records out

80 bytes (80 B) copied, 0.000151458 s, 528 kB/s

0+1 records in

0+1 records out

79 bytes (79 B) copied, 0.000120032 s, 658 kB/s

SELECT \* from bc.bccopy

\*

ERROR at line 1:

ORA-01578: ORACLE data block corrupted (file # 18, block # 129) ORA-01110: data file 18: '/u01/backup/orclcdb/pdb1/bctbs01.dbf'

…

$

1. Using RMAN as SYSBACKUP, connect to the orclcdb instance and check for failures with the LIST FAILURE command.

$ **rman target "'/ as sysbackup'"**

…

connected to target database: ORCLCDB (DBID=2778750799)

RMAN> **LIST FAILURE;**

using target database control file instead of recovery catalog Database Role: PRIMARY

List of Database Failures

=========================

Failure ID Priority Status

Time Detected

Summary

1889

HIGH

OPEN

01-JUL-19

Datafile 179:

'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bctbs01.dbf' contains one or more corrupt blocks

RMAN>

1. Use the RMAN ADVISE FAILURE command and view the suggested repair strategy.

RMAN> **ADVISE FAILURE;**

Database Role: PRIMARY

List of Database Failures

=========================

Failure ID Priority Status

Time Detected Summary

1889

HIGH

OPEN

01-JUL-19

Datafile 179:

'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bctbs01.dbf' contains one or more corrupt blocks

analyzing automatic repair options; this may take some time allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=275 device type=DISK analyzing automatic repair options complete

Mandatory Manual Actions

========================

no manual actions available

Optional Manual Actions

=======================

no manual actions available

Automated Repair Options

========================

Option Repair Description

1 Recover multiple corrupt blocks in datafile 179 Strategy: The repair includes complete media recovery with no

data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3585523541.hm

RMAN>

1. Use the RMAN REPAIR FAILURE command to recover the corrupted blocks.

RMAN> **REPAIR FAILURE;**

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3585523541.hm

contents of repair script:

# block media recovery for multiple blocks recover datafile 179 block 129 to 130;

Do you really want to execute the above repair (enter YES or NO)? **YES**

executing repair script

Starting recover at 01-JUL-19 using channel ORA\_DISK\_1

channel ORA\_DISK\_1: restoring block(s) from datafile copy

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8C28E6F854EB7DBBE0536 210ED0AFDD9/datafile/o1\_mf\_bctbs\_gkn2jj8m\_.dbf

starting media recovery

media recovery complete, elapsed time: 00:00:03

Finished recover at 01-JUL-19 repair failure complete

RMAN>

1. Optionally, confirm that there is no other failure.

RMAN> **list failure;**

Database Role: PRIMARY

no failures found that match specification RMAN>

1. Exit RMAN.

RMAN> **exit**

…

**Note:** This script may end with an error showing that data file system01.dbf cannot be opened.

1. Verify that the corrupt block has been recovered by running a full table scan on

bc.bccopy.

$ **sqlplus / as sysdba**

…

SQL> **alter session set container = orclpdb1;**

Session altered.

SQL> **SELECT \* FROM BC.BCCOPY;**

…

SQL> **EXIT**

1. Clean up from the block corruption practice by executing the cleanup\_05\_01.sh script. You can view the script output in the /tmp/cleanup.log file.

$ **./cleanup\_05\_01.sh**

$

1. Keep all terminal windows open for the next practice.

# Practices for Lesson 18:

# Performing Additional Recovery Operations

## Practices for Lesson 18: Overview

### Overview

In these practices, you will recover from a number of different database failures. It is highly recommended to complete those that you start, because earlier ones affect practices that follow.

In preparation for the next practice, back up your orclcdb database,

and ensure that no failures are listed.

$ **rman target /**

RMAN> **BACKUP DATABASE;**

…

RMAN> **list failure;**

Database Role: PRIMARY

no failures found that match specification RMAN> **exit**

…

## Practice 18-1: Restoring the Control File

### Overview

In this practice, you first create an issue for recovery by deleting the control02.ctl control file. After creating the issue, you *must* restore this single "lost" control file.

### Assumptions

A full backup of the database is available.

You have two terminal windows open in which you are logged in as the oracle OS user,

$HOME/labs/DBMod\_Recovery is the current directory, and environment variables point to the orclcdb database instance.

### Tasks

1. Prepare for this practice by executing the setup\_06\_02.sh script from the

$HOME/labs/DBMod\_Recovery directory. This script creates a new tablespace and user. As the new user, the script creates a table and populates it. The script creates a backup of the tablespace, and then updates the table.

The script saves its output in the /tmp/setup.log file. You can view this log file while the script executes.

$ **./setup\_06\_02.sh**

$

1. Cause a failure in the database by executing the break\_06\_02.sh script from the

$HOME/labs/DBMod\_Recovery directory. The script saves its output in the

/tmp/break.log file.

$ **./break\_06\_02.sh**

$

1. Attempt to start the database. Review the error messages. Then exit SQL\*Plus.

$ **sqlplus / as sysdba**

…

Connected to an idle instance.

SQL> **startup**

ORACLE instance started.

…

ORA-00205: error in identifying control file, check alert log for more info

SQL> **exit**

…

$

1. Use adrci to view the orclcdb/orclcdb alert log. Scroll to the most recent entries to see the errors from this practice. The list of diagnostic homes may vary. Select the home for orclcdb/orclcdb (*option 2* in this example). Your option number may be different.

$ **adrci**

…

adrci> **set editor gedit**

adrci> **show alert**

Choose the home from which to view the alert log:

1: diag/rdbms/dummy/orclcdb

2: diag/rdbms/orcldg/orcldg

2: diag/rdbms/**orclcdb**/**orclcdb**

4: diag/rdbms/ORCLOEM/ORCLOEM

5: diag/tnslsnr/edvmr1p0/listener Q: to quit

Please select option: **3** (Choose according to your environment)

/\* Scroll to bottom of the file in gedit to see the following \*/

…

ALTER DATABASE MOUNT

ORA-00210: cannot open the specified control file ORA-00202: control file:

'/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl' ORA-27037: unable to obtain file status

Linux-x86\_64 Error: 2: No such file or directory Additional information: 7

ORA-205 signalled during: ALTER DATABASE MOUNT...

Errors in file

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/trace/orclcdb\_mz00\_76 15.trc:

ORA-00202: control file: '/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl'

ORA-27037: unable to obtain file status

Linux-x86\_64 Error: 2: No such file or directory Additional information: 7

2019-07-01 17:28:25.169000 +00:00

Checker run found 1 new persistent data failures 2018-07-25 19:52:00.135000 +00:00

1. Close the gedit window by clicking the close window icon (x) and then exit adrci.

Please select option: **Q**

adrci> **exit**

$

1. Check the Data Recovery Advisor for a solution. Preview the suggested solution before executing it.
   1. Log in to RMAN and run the LIST FAILURE command.

$ **rman target "'/ as sysbackup'"**

…

connected to target database: ORCLCDB (not mounted) RMAN> **LIST FAILURE;**

using target database control file instead of recovery catalog List of Database Failures

=========================

Failure ID Priority Status

Time Detected

Summary

2312

CRITICAL OPEN

01-JUL-19

Control file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl is missing

RMAN>

* 1. Run the ADVISE FAILURE command.

RMAN> **ADVISE FAILURE;**

List of Database Failures

=========================

2312

CRITICAL OPEN

01-JUL-19

Control file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl is missing

analyzing automatic repair options; this may take some time allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=255 device type=DISK analyzing automatic repair options complete

Mandatory Manual Actions

========================

no manual actions available

Optional Manual Actions

=======================

no manual actions available

Automated Repair Options

========================

Option Repair Description

1

Use a multiplexed copy to restore control file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3850033214.hm

RMAN>

* 1. Run the REPAIR FAILURE PREVIEW command.

RMAN> **REPAIR FAILURE PREVIEW;**

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3850033214.hm

contents of repair script:

# restore control file using multiplexed copy

restore controlfile from '/u01/app/oracle/oradata/ORCLCDB/control01.ctl';

sql 'alter database mount';

RMAN>

1. Restore the control file. You can either execute the commands via the RMAN command line, or use the REPAIR FAILURE command to perform the task for you.

**Note:** Any existing copy of the control file can be used to restore the missing copy. Enter Y or YES when prompted to execute the repair and to open the database.

RMAN> **REPAIR FAILURE;**

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_3850033214.hm

contents of repair script:

# restore control file using multiplexed copy restore controlfile from

'/u01/app/oracle/oradata/ORCLCDB/control01.ctl'; sql 'alter database mount';

Do you really want to execute the above repair (enter YES or NO)? **YES**

executing repair script

Starting restore at 01-JUL-19 using channel ORA\_DISK\_1

channel ORA\_DISK\_1: copied control file copy

output file name=/u01/app/oracle/oradata/ORCLCDB/control01.ctl

output file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl

Finished restore at 01-JUL-19

sql statement: alter database mount released channel: ORA\_DISK\_1 repair failure complete

Do you want to open the database (enter YES or NO)? **YES**

database opened

RMAN>

1. Use the LIST FAILURE command to verify that the failure has been repaired.

RMAN> **list failure;**

Database Role: PRIMARY

no failures found that match specification RMAN>

1. In your second terminal window, clean up the practice environment by running the

cleanup\_06\_02.sh script. The script saves its output in the /tmp/cleanup.log file.

$ **./cleanup\_06\_02.sh**

$

1. Return to the RMAN terminal session. In preparation for the next practice, back up your

orclcdb database, remove obsolete backups, and ensure that no failures are listed.

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

…

RMAN> **delete noprompt obsolete;**

…

RMAN> **list failure;**

Database Role: PRIMARY

no failures found that match specification RMAN> **exit**

…

$

## Practice 18-3: Recovering from the Loss of All Control Files

### Overview

In this practice, you create an issue by removing control files. After creating the issue, you must restore the control files.

### Assumptions

A full backup of the database is available. Autobackup of the control file and SPFILE is configured.

You have two terminal windows open in which you are logged in as the oracle OS user,

$HOME/labs/DBMod\_Recovery is the current directory, and environment variables point to the orclcdb database instance.

### Tasks

1. Prepare for this practice by executing the setup\_06\_03.sh script from the

$HOME/labs/DBMod\_Recovery directory. This script creates a new tablespace and user. As the new user, the script creates a table and populates it. The script creates a backup of the tablespace, and then updates the table.

The script saves its output in the /tmp/setup.log file. You can view this log file while the script executes.

$ **./setup\_06\_03.sh**

$

1. Cause a failure in the database by executing the break\_06\_03.sh script from the

$HOME/labs/DBMod\_Recovery directory. The script saves its output in the

/tmp/break.log file.

$ **./break\_06\_03.sh**

$

1. Log in to SQL\*Plus and attempt to start the database. Notice the error message. Exit from SQL\*Plus.

$ **sqlplus / as sysdba**

…

Connected to an idle instance.

SQL> **startup**

ORACLE instance started.

…

ORA-00205: error in identifying control file, check alert log for more info

SQL> **exit**

1. View the alert log (*option* 2 in this example). Your option number may be different. Scroll to the most recent entries to see the errors from this practice.

$ **adrci**

…

ADR base = "/u01/app/oracle" adrci> **set editor gedit** adrci> **show alert**

Choose the home from which to view the alert log:

1: diag/rdbms/dummy/orclcdb

1: diag/rdbms/orcldg/

orcldg

3: diag/rdbms/**orclcdb**/**orclcdb**

3: diag/rdbms/ubff\_pitr\_orclpdb1\_orclcdb/uBFF

4: diag/rdbms/ORCLOEM/ORCLOEM

5: diag/tnslsnr/edvmr1p0/listener Q: to quit

Please select option: **3** (Choose as per your environment)

/\* Scroll to bottom of the file in gedit to see the following \*/

…

ORA-00202: control file: '/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl'

ORA-27037: unable to obtain file status

ORA-00210: cannot open the specified control file

ORA-00202: control file: '/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl'

ORA-27037: unable to obtain file status

Linux-x86\_64 Error: 2: No such file or directory Additional information: 7

ORA-00210: cannot open the specified control file

ORA-00202: control file: '/u01/app/oracle/oradata/ORCLCDB/control01.ctl'

ORA-27037: unable to obtain file status

Linux-x86\_64 Error: 2: No such file or directory Additional information: 7

Checker run found 3 new persistent data failures

**Note:** There are three failures, all control files are missing.

1. Exit from viewing the alert log by clicking the "x" icon to close the gedit window. Exit

adrci by entering Q, and then exit.

…

Please select option: **Q**

adrci> **exit**

$

1. Use the RMAN LIST FAILURE and ADVISE FAILURE commands to determine the failures and proposed solutions.

#### $ rman target "'/ as sysbackup'"

…

connected to target database: ORCLCDB (not mounted)

RMAN> **list failure;**

using target database control file instead of recovery catalog List of Database Failures

=========================

Failure ID Priority Status Time Detected Summary

4030 CRITICAL OPEN 02-JUL-19 Control file

/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl is missing

4035 CRITICAL OPEN 02-JUL-19 Control file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl is missing

4032 CRITICAL OPEN 02-JUL-19 Control file

/u01/app/oracle/oradata/ORCLCDB/control01.ctl is missing

RMAN> **advise failure;**

List of Database Failures

=========================

Failure ID Priority Status Time Detected Summary

4030 CRITICAL OPEN 02-JUL-19 Control file

/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl is missing

4035 CRITICAL OPEN 02-JUL-19 Control file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl is missing

4032 CRITICAL OPEN 02-JUL-19 Control file

/u01/app/oracle/oradata/ORCLCDB/control01.ctl is missing

analyzing automatic repair options; this may take some time allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=237 device type=DISK analyzing automatic repair options complete

Mandatory Manual Actions

========================

no manual actions available

Optional Manual Actions

=======================

* 1. If file

/u01/app/oracle/controlfiles\_dir/ORCLCDB/control03.ctl was unintentionally renamed or moved, restore it

* 1. If file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl was unintentionally renamed or moved, restore it

* 1. If file /u01/app/oracle/oradata/ORCLCDB/control01.ctl was unintentionally renamed or moved, restore it
  2. If this is a standby database, restore the controlfile for a standby database using RESTORE STANDBY CONTROLFILE FROM AUTOBACKUP command
  3. If this is a primary database and a standby database is available, then perform a Data Guard failover initiated from the standby

Automated Repair Options

========================

Option Repair Description

1 Restore a backup control file

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_1494177699.hm

RMAN>

1. Review the commands generated by the REPAIR FAILURE PREVIEW command.

RMAN> **repair failure preview;**

Strategy: The repair includes complete media recovery with no data loss

Repair script:

/u01/app/oracle/diag/rdbms/orclcdb/orclcdb/hm/reco\_1494177699.hm

contents of repair script: # restore control file

restore controlfile from autobackup; sql 'alter database mount';

RMAN>

1. One option now would be to use the automated recovery command REPAIR FAILURE. However, doing so introduces new failures, such that you would subsequently need to run these RMAN commands in a series, repeated as needed to correct the failures:

LIST FAILURE;

ADVISE FAILURE;

REPAIR FAILURE PREVIEW; REPAIR FAILURE;

So to avoid going through the above steps repeatedly with the Data Recovery Advisor, follow steps 8 through 11 of this practice exactly as described.

Use the RMAN command line to restore the control files and mount the database.

RMAN> **restore controlfile from autobackup;**

Starting restore at 02-JUL-19 using channel ORA\_DISK\_1

recovery area destination: /u01/app/oracle/fast\_recovery\_area database name (or database unique name) used for search: ORCLCDB

channel ORA\_DISK\_1: AUTOBACKUP

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_07\_01

/o1\_mf\_s\_1012509894\_gknw675v\_.bkp found in the recovery area

AUTOBACKUP search with format "%F" not attempted because DBID was not set

channel ORA\_DISK\_1: restoring control file from AUTOBACKUP

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_07\_01

/o1\_mf\_s\_1012509894\_gknw675v\_.bkp

channel ORA\_DISK\_1: control file restore from AUTOBACKUP complete

output file name=/u01/app/oracle/oradata/ORCLCDB/control01.ctl

output file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/control02.ctl

Finished restore at 02-JUL-19 RMAN> **ALTER DATABASE MOUNT;**

released channel: ORA\_DISK\_1 Statement processed

RMAN>

1. Attempt to open the database.

RMAN> **ALTER DATABASE OPEN;**

RMAN-00571: ==================================================== RMAN-00569: =========== ERROR MESSAGE STACK FOLLOWS ============ RMAN-00571: ====================================================

RMAN-03002: failure of sql statement command at 07/02/2019 14:48:28

ORA-01589: must use RESETLOGS or NORESETLOGS option for database open

RMAN>

**Question:** Why is RESETLOGS necessary?

**Answer:** RESETLOGS is required because the SCN in the restored control file does not match the SCN recorded in the data files.

1. Attempt to open the database with the RESETLOGS option.

RMAN> **ALTER DATABASE OPEN RESETLOGS;**

RMAN-00571: ==================================================== RMAN-00569: =========== ERROR MESSAGE STACK FOLLOWS ============ RMAN-00571: ====================================================

RMAN-03002: failure of sql statement command at 07/02/2019 14:48:39

ORA-01194: file 1 needs more recovery to be consistent ORA-01110: data file 1: '/u01/app/oracle/oradata/ORCLCDB/system01.dbf'

RMAN>

**Question:** Why did the database open still fail with the RESETLOGS option?

**Answer:** The SCN in the control file is older than the SCN in the data files and the data files have not been restored with the UNTIL cause. The database needs to be recovered so that the control file can be synchronized with the data files.

1. Recover the database.

RMAN> **recover database;**

Starting recover at 02-JUL-19

Starting implicit crosscheck backup at 02-JUL-19 allocated channel: ORA\_DISK\_1

channel ORA\_DISK\_1: SID=258 device type=DISK Crosschecked 12 objects

Finished implicit crosscheck backup at 02-JUL-19

Starting implicit crosscheck copy at 02-JUL-19 using channel ORA\_DISK\_1

Finished implicit crosscheck copy at 02-JUL-19

searching for all files in the recovery area cataloging files...

cataloging done

List of Cataloged Files

=======================

File Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/autobackup/2019\_07\_01

/o1\_mf\_s\_1012509894\_gknw675v\_.bkp File Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_07\_02

/o1\_mf\_1\_6\_gkp095or\_.arc File Name:

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/8C28E6F854EB7DBBE0536

210ED0AFDD9/datafile/o1\_mf\_bartbs\_gkpv2ngh\_.dbf

using channel ORA\_DISK\_1

starting media recovery

archived log for thread 1 with sequence 6 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_07\_02

/o1\_mf\_1\_6\_gkp095or\_.arc

archived log for thread 1 with sequence 7 is already on disk as file

/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo1a.log archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/archivelog/2019\_

07\_02/o1\_mf\_1\_6\_gkp095or\_.arc thread=1 sequence=6

archived log file name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo1a

.log thread=1 sequence=7

creating datafile file number=189 name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf archived log file

name=/u01/app/oracle/fast\_recovery\_area/ORCLCDB/onlinelog/redo1a

.log thread=1 sequence=7

media recovery complete, elapsed time: 00:00:01 Finished recover at 02-JUL-19

RMAN>

1. Open the database with RESETLOGS.

RMAN> **ALTER DATABASE OPEN RESETLOGS;**

Statement processed

RMAN> **ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN;**

Statement processed

1. Use the LIST FAILURE command to verify that the failure has been repaired.

RMAN> **list failure;**

Database Role: PRIMARY

no failures found that match specification RMAN>

1. In your second terminal window, optionally log in to SQL\*Plus and query V$DATABASE to view the values of DBID and CURRENT\_SCN of both orclcdb and orclpdb1. Exit SQL\*Plus when finished.

$ **sqlplus / as sysdba**

…

SQL> **SELECT NAME, DBID, CURRENT\_SCN, LOG\_MODE, OPEN\_MODE FROM V$DATABASE;**

NAME

DBID CURRENT\_SCN LOG\_MODE

OPEN\_MODE

ORCLCDB 2778750799

11419045 ARCHIVELOG READ WRITE

SQL> **alter session set container=orclpdb1;**

Session altered.

SQL> **SELECT NAME, DBID, CURRENT\_SCN, LOG\_MODE, OPEN\_MODE FROM V$DATABASE;**

1. Clean up the practice environment by using the cleanup\_06\_03.sh script from the

NAME

DBID CURRENT\_SCN LOG\_MODE

OPEN\_MODE

ORCLCDB 2778750799

11419154 ARCHIVELOG

READ WRITE

SQL> **exit**

…

$

$HOME/labs/DBMod\_Recovery directory. The script saves its output in the

/tmp/cleanup.log file.

$ **./cleanup\_06\_03.sh**

$

1. Return to the RMAN terminal session. In preparation for the next practice, back up your orclcdb database, remove obsolete backups, and ensure that no failures are listed. Then exit from RMAN.

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

…

RMAN> **delete noprompt obsolete;**

…

RMAN> **list failure;**

Database Role: PRIMARY

no failures found that match specification RMAN> **exit**

…

$

**Practice 18-4: Restoring the Password File**

### Overview

In this practice, you recover from the loss of the database password file. The database password is required for remote access to the database by SYSDBA privileged users.

### Assumptions

A full backup of the database is available.

You have two terminal windows open in which you are logged in as the oracle OS user,

$HOME/labs/DBMod\_Recovery is the current directory, and environment variables point to the orclcdb database instance.

### Tasks

1. Cause a failure in the database by executing the break\_06\_04.sh script from the

$HOME/labs/DBMod\_Recovery directory. The script saves its output in the

/tmp/break.log file. You can view this file while and after the script executes.

$ **./break\_06\_04.sh**

$

1. Attempt to connect to the database using a remote connection. Notice the error messages. Refer to the “*Course Practice Environment: Security Credentials*” document for the correct password.

$ **sqlplus sys@orclcdb as sysdba**

…

Enter password: **\*\*\*\*\*\*\*\*\***

ERROR:

ORA-01017: invalid username/password; logon denied

Enter user-name: **^C**

$

**Note:** The remote connection requires the use of a password file.

1. Check if the password file exists. The name of the orclcdb database password file for Linux and UNIX systems is $ORACLE\_HOME/dbs/orapworclcdb.

$ **ls $ORACLE\_HOME/dbs/orapw\***

The orapworclcdb.ora should not be listed because it was deleted by the

break\_06\_04.sh script. But files for other databases will exist, ignore it.

**Note:** It is critically important to the security of your system that you protect your password file and the environment variables that identify the location of the password file. Any user with access to these could potentially compromise the security of the connection.

1. Optionally, view the description of the orapwd parameters. Invoke orapwd in a terminal window.

$ **orapwd**

Usage: orapwd file=<fname> force=<y/n> asm=<y/n> dbuniquename=<dbname> format=<12/12.2> delete=<y/n> input\_file=<input-fname> sys=<y/password/external(<sys-external-name>)>

sysbackup=<y/password/external(<sysbackup-external-name>)> sysdg=<y/password/external(<sysdg-external-name>)> syskm=<y/password/external(<syskm-external-name>)>

Usage: orapwd describe file=<fname>

where

file - name of password file (required), password - password for SYS will be prompted

if not specified at command line. Ignored, if input\_file is specified,

force - whether to overwrite existing file (optional), asm - indicates that the password to be stored in

Automatic Storage Management (ASM) disk group is an ASM password. (optional),

dbuniquename - unique database name used to identify database password files residing in ASM diskgroup only. Ignored when asm option is specified (optional),

format - use format=12 for new 12c features like SYSBACKUP, SYSDG

and

SYSKM support, longer identifiers, SHA2 Verifiers etc. use format=12.2 for 12.2 features like enforcing user profile (password limits and password complexity) and account status for administrative users.

If not specified, format=12.2 is default (optional), delete - drops a password file. Must specify 'asm',

'dbuniquename' or 'file'. If 'file' is specified,

the file must be located on an ASM diskgroup (optional), input\_file - name of input password file, from where old user

entries will be migrated (optional),

sys - specifies if SYS user is password or externally

authenticated.

For external SYS, also specifies external name. SYS={y/password} specifies if SYS user password needs to be changed when used with input\_file,

sysbackup - creates SYSBACKUP entry (optional).

Specifies if SYSBACKUP user is password or externally authenticated. For external SYSBACKUP, also specifies

external name. Ignored, if input\_file is specified, sysdg - creates SYSDG entry (optional).

Specifies if SYSDG user is password or externally authenticated.

For external SYSDG, also specifies external name. Ignored, if input\_file is specified,

syskm - creates SYSKM entry (optional).

Specifies if SYSKM user is password or externally authenticated.

For external SYSKM, also specifies external name. Ignored, if input\_file is specified,

describe - describes the properties of specified password file (required).

There must be no spaces around the equal-to (=) character.

$

1. Create a new password file by using the orapwd utility. Use the password specified for this step in the “*Course Practice Environment: Security Credentials*” document. Replace

*fenago* with the correct password.

$ **orapwd FILE=$ORACLE\_HOME/dbs/orapworclcdb ENTRIES=15 force=Y password=fenago format=12**

**Note:** When you exceed the allocated number of password entries, you must create a new password file. To avoid this necessity, allocate more entries than you think you will ever need.

1. Test the remote SYSDBA login. Now it should be successful. Use the password specified for this step in the “*Course Practice Environment: Security Credentials*” document. Replace

*fenago* with the correct password.

$ **sqlplus sys@orclcdb as sysdba**

…

Enter password: ***<password>*** <<< Not displayed

… SQL>

1. Optionally, review the V$PWFILE\_USERS view.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SQL> **desc** | **V$PWFILE\_USERS** |  | | | |
| Name |  |  | Null? |  | Type |
| USERNAME SYSDBA SYSOPER  SYSASM |  |  |  |  | VARCHAR2(128) VARCHAR2(5) VARCHAR2(5)  VARCHAR2(5) |

SYSBACKUP SYSDG SYSKM

ACCOUNT\_STATUS PASSWORD\_PROFILE LAST\_LOGIN LOCK\_DATE EXPIRY\_DATE EXTERNAL\_NAME AUTHENTICATION\_TYPE COMMON

CON\_ID

VARCHAR2(5) VARCHAR2(5) VARCHAR2(5) VARCHAR2(30) VARCHAR2(128)

TIMESTAMP(9) WITH TIME ZONE DATE

DATE VARCHAR2(1024) VARCHAR2(8) VARCHAR2(3)

NUMBER

SQL> **set pagesize 100**

SQL> **SELECT \* FROM V$PWFILE\_USERS;**

USERNAME

SYSDB SYSOP SYSAS SYSBA SYSDG SYSKM ACCOUNT\_STATUS PASSWORD\_PROFILE

LAST\_LOGIN

LOCK\_DATE

EXPIRY\_DATE

EXTERNAL\_NAME

AUTHENTI COM

CON\_ID

SYS

TRUE TRUE FALSE FALSE FALSE FALSE OPEN

26-JUL-18 02.14.37.000000000 PM +00:00

PASSWORD YES

0

SQL>

1. Exit SQL\*Plus.

SQL> **exit**

…

$

1. Close all terminal windows open.

**Practices for Lesson 19: Oracle Flashback Technology Overview**

**Practices for Lesson 19**

There are no practices for Lesson 19.

**Practices for Lesson 20: Using Logical Flashback Features**

**Practices for Lesson 20: Overview**

### Overview

In these practices, you will configure your database for the use of flashback technologies. You will then use flashback technologies to restore a dropped table and reverse the actions of a transaction.

**Practice 20-1: Preparing to Use Flashback Technologies**

### Overview

In this practice, you will configure the database to use the flashback transaction and flashback table features.

### Assumptions

You have two terminal windows open in which you are logged in as the oracle OS user and

/home/oracle/labs/DBMod\_Flashback is the current directory. Each terminal window is connected to the orclpdb1 instance.

### Tasks

1. Open a terminal window. Use SQL\*Plus to connect to ORCLPDB1 as the SYS user.

$ **sqlplus sys/fenago as sysdba**

SQL> **alter session set container=orclpdb1;**

session set container=orclpdb1;

1. Determine how far the undo data will allow you to flash back transactions in the database. The V$UNDOSTAT view contains up to four days of statistics, over each 10-minute interval. The first row contains the current (partial) time period statistics. (*Your value may be different.*)

SQL> **select tuned\_undoretention from v$undostat where rownum = 1;**

TUNED\_UNDORETENTION

900

SQL>

**Question:** What does the value of the TUNED\_UNDORETENTION time represent? **Answer:** The number of seconds the data is being retained in the database at any given time. Undo retention is not guaranteed by default. If the system needs more space, the

Oracle Database server can overwrite unexpired undo with more recently generated undo

data.

1. Set the UNDO\_RETENTION parameter and RETENTION GUARANTEE clause on the undo tablespace to guarantee retention for 24 hours. Change the properties of the tablespace so you will not run out of space in the tablespace.
   1. Change the UNDO\_RETENTION parameter to 14400 seconds (4 hours).

**Note:** Increasing the value of UNDO\_RETENTION to more than a few days can cause unreasonable growth of the undo tablespace.

SQL> **ALTER SYSTEM SET UNDO\_RETENTION = 14400 SCOPE=BOTH;**

System altered.

SQL>

* 1. Determine the name of the undo tablespace.
  2. Change the RETENTION GUARANTEE value of the undo tablespace.

SQL> **ALTER TABLESPACE UNDOTBS1 RETENTION GUARANTEE;**

Tablespace altered. SQL>

SQL> **SHOW PARAMETER UNDO**

NAME

TYPE

VALUE

temp\_undo\_enabled undo\_management undo\_retention undo\_tablespace

SQL>

boolean string integer

string

FALSE AUTO 14400

UNDOTBS1

* 1. Find the names of the data files associated with the UNDOTBS1 tablespace. Note the

FILE\_ID value.

SQL> **select file\_name, file\_id from dba\_data\_files where tablespace\_name like '%UNDO%';**

FILE\_NAME FILE\_ID

/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf 11 SQL>

* 1. Configure the undo tablespace data file to automatically extend if more space is needed to keep unexpired undo and active undo records. Use your own FILE\_ID value from previous command instead of UPDATE\_HERE.

SQL> **ALTER DATABASE DATAFILE 'UPDATE\_HERE' AUTOEXTEND ON MAXSIZE UNLIMITED;**

Database altered. SQL>

****

**Question:** What happens if undo retention is guaranteed and no more space is available for active undo records (either because the undo tablespace has filled, reached the maximum size, or there is no more space left on the storage device [disk])?

**Answer:** Transactions fail due to lack of space in the undo tablespace.

1. View the value of the RECYCLEBIN parameter, then exit SQL\*Plus.

**Note:** ON is the default value.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| SQL> **show parameter** | **recyclebin** |  | | | |
| NAME |  |  | TYPE |  | VALUE |
| recyclebin |  |  | string |  | on |
| SQL> **exit** |  |  |  |  |  |
| …  $ |  |  |  |  |  |

**Practice 20-2: Restoring a Dropped Table**

### Overview

In this practice, you will recover a table that has been dropped.

### Assumptions

The RECYCLEBIN parameter is set to ON (which you confirmed in Practice 2-1 Step 4).

You have two terminal windows open in which you are logged in as the oracle OS user and

/home/oracle/labs/DBMod\_Flashback is the current directory. Each terminal window is connected to the orclpdb1 instance.

### Tasks

1. Execute the setup\_02\_02.sh script to create the practice environment. The script saves its output in the /home/oracle/labs/DBMod\_Flashback/setup.log file.

$ **cd /home/oracle/labs/DBMod\_Flashback**

$ **./setup\_02\_02.sh**

$

1. Execute the break\_02\_02.sh script to simulate the work done by a developer. The script saves its output in the /home/oracle/labs/DBMod\_Flashback/break.log file.

$ **./break\_02\_02.sh**

$

1. A developer, who has been using the BAR Oracle user account comes to you and asks you to restore a table that was dropped. This table has had several iterations but the one the developer needs was named BAR102 in the BAR schema. It should have 12 columns, one of which was named LOCATION\_ID. There is currently a BAR102 table in the BAR schema. Restore the requested table to BAR102A.
   1. Attempt to use the SHOW RECYCLEBIN command to view the contents of the recycle bin for orclpdb1.

$ **sqlplus sys/fenago as sysdba**

SQL> **ALTER PLUGGABLE DATABASE ALL OPEN;**

SQL> **alter session set container=orclpdb1;**

SQL> **show recyclebin**

SQL>

**Note:** The SHOW RECYCLEBIN command shows only those objects that belong to the *current* user. Because you are logged in as the SYS user, the SHOW RECYCLEBIN command does not show the dropped tables you are interested in restoring.

* 1. Examine the objects in the DBA\_RECYCLEBIN view. Optionally, change the SQL\*Plus page size to 99 lines.

SQL> **set pages 99**

SQL> **select original\_name, object\_name, droptime from dba\_recyclebin where owner ='BAR';**

ORIGINAL\_NAME OBJECT\_NAME DROPTIME

BAR102

BIN$jNKlrSHcInLgU6YR7Qqg/Q==$0 2019-07-04:02:22:19

BAR102

BIN$jNKlrSHhInLgU6YR7Qqg/Q==$0 2019-07-04:02:22:19

BAR101

BIN$jNKlrSHmInLgU6YR7Qqg/Q==$0 2019-07-04:02:22:19

**Note: Your lab environment will have different object names.**

**Note:** In the preceding output, you see the same object dropped twice at different points in time. With the time stamp, you determine which version of the table you really want to restore.

* 1. Determine which object contains the column of interest. *Your object names will be different*. *Use your own values from the previous step*.

**Hint:** Select the column from the tables, the one which does not report an error is the one you want.

**Note:** The object name from the recycle bin must be in double quotation marks because it may contain special chars (update objectname in below commands)

SQL> **SELECT location\_id**

**FROM BAR.BIN$jNKlrSHcInLgU6YR7Qqg/Q==$0**

**WHERE rownum = 1;**

Select location\_id

\*

ERROR at line 1:

ORA-00904: "LOCATION\_ID": invalid identifier

SQL> **SELECT location\_id**

**FROM BAR."BIN$jNKlrSHhInLgU6YR7Qqg/Q==$0"**

**where rownum = 1;**

LOCATION\_ID SQL>

* 1. Restore the object that has the correct columns. *Use the correct object name that you found in the previous steps*.

SQL> **FLASHBACK TABLE BAR."BIN$jNKlrSHhInLgU6YR7Qqg/Q==$0" TO BEFORE DROP RENAME TO BAR102A;**

Flashback complete. SQL>

1. Confirm that the BAR.BAR102A table has been restored by selecting the first row. Then exit SQL\*Plus.

**Note**: It does not matter which row is displayed from the query.

SQL> **select \* from BAR.BAR102A where rownum = 1;**

EMPLOYEE\_ID FIRST\_NAME

LAST\_NAME

EMAIL

SALARY

PHONE\_NUMBER

HIRE\_DATE JOB\_ID

COMMISSION\_PCT MANAGER\_ID DEPARTMENT\_ID LOCATION\_ID

100 Steven

SKING 24001

King

515.123.4567

17-JUN-03 AD\_PRES

90

SQL> **exit**

$

1. Clean up from this practice by executing the cleanup\_02\_02.sh script.

**Note:** This script uses the PURGE DBA\_RECYCLEBIN command to remove all objects from the recycle bin. The script saves its output in the

/home/oracle/labs/DBMod\_Flashback/cleanup.log file.

$ **./cleanup\_02\_02.sh**

$

**Practice 20-3: Using Flashback Table**

### Overview

In this practice, you will use flashback table to reverse a transaction.

### Assumptions

Practice 2-1 has been completed.

You have two terminal windows open in which you are logged in as the oracle OS user and

/home/oracle/labs/DBMod\_Flashback is the current directory. Each terminal window is connected to the orclpdb1 instance.

### Tasks

1. Execute the setup\_02\_03.sh script to create the user and tables used in this practice. The tables have a foreign-key relationship. The script saves its output in the

/home/oracle/labs/DBMod\_Flashback/setup.log file.

$ **./setup\_02\_03.sh**

$

1. Determine the current time to the nearest second. Record this as T1.

**Note:** Your date will be different.

$ **sqlplus sys/fenago as sysdba**

SQL> **alter session set container=orclpdb1;**

SQL> **select to\_char(sysdate, 'yyyy-mm-dd:hh24:mi:ss') from dual;**

SYSDATE

2019-07-04:03:00:25

SQL>

1. Open a new terminal window and execute the break\_02\_03.sh script. This simulates a transaction that scrambles the data in the BARCOPY and BARDEPT tables. There is a foreign key constraint between BARCOPY and BARDEPT. The script saves its output in the

/home/oracle/labs/DBMod\_Flashback/break.log file.

$ **cd /home/oracle/labs/DBMod\_Flashback**

$ **./break\_02\_03.sh**

$

1. Return to the SQL\*Plus terminal session connected to ORCLPDB1 as the SYS user. The HR representative reports that an employee has incorrectly changed the names of the departments, and scrambled which employees are assigned to which departments. The tables were correct at time T1 and no authorized changes have been made since that time. The tables involved are BAR.BARCOPY and BAR.DEPT. Restore the tables to the state they were at T1 (determined in step 2).

Because there is a foreign-key relationship, both tables must be restored. Continue in your existing SQL\*Plus session.

SQL> **ALTER TABLE BAR.BARDEPT ENABLE ROW MOVEMENT;**

Table altered.

SQL> **ALTER TABLE BAR.BARCOPY ENABLE ROW MOVEMENT;**

Table altered.

SQL> **FLASHBACK TABLE BAR.BARDEPT TO TIMESTAMP TO\_TIMESTAMP('2019-07-04:03:00:25','YYYY-MM-DD:HH24:MI:SS');**

Flashback complete.

SQL> **FLASHBACK TABLE BAR.BARCOPY TO TIMESTAMP TO\_TIMESTAMP('2019-07-04:03:00:25','YYYY-MM-DD:HH24:MI:SS');**

Flashback complete. SQL>

1. Check that you have correctly restored the tables. The results of the following query should match yours, although the row order may be different. Then exit SQL\*Plus.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SQL> **@check\_02\_03.sql** | | | | |
| DEPARTMENT\_ID |  | DEPARTMENT\_NAME |  | COUNT(\*) |
| 10 |  | Administration |  | 4 |
| 20 |  | Marketing |  | 8 |
| 30 |  | Purchasing |  | 24 |
| 40 |  | Human Resources |  | 4 |
| 50 |  | Shipping |  | 180 |
| 60 |  | IT |  | 20 |
| 70 |  | Public Relations |  | 4 |
| 80 |  | Sales |  | 136 |
| 90 |  | Executive |  | 12 |

100 Finance

110 Accounting

24

8

11 rows selected.

SQL> **exit**

…

$

1. Clean up the practice environment by running the cleanup\_02\_03.sh script.

$ **./cleanup\_02\_03.sh**

$

**Practices for Lesson 21:**

**Using Flashback Database**

**Practices for Lesson 21: Overview**

### Overview

In these practices, you will enable flashback logging and perform flashback database.

**Practice 21-1: Enabling Flashback Logging**

### Overview

In this practice, you will enable flashback logging.

### Assumptions

You have two terminal windows open in which you are logged in as the oracle OS user and

/home/oracle/labs/DBMod\_Flashback is the current directory. Each terminal window is connected to the orclpdb1 instance.

### Tasks

1. Determine whether flashback logging is enabled while still connected to the container database.

$ **sqlplus / as sysdba**

…

SQL> **SELECT flashback\_on FROM v$database;**

FLASHBACK\_ON NO

SQL>

1. Enable flashback logging and verify.

SQL> **ALTER DATABASE FLASHBACK ON;**

Database altered.

SQL> **select FLASHBACK\_ON from v$database;**

FLASHBACK\_ON

YES

1. In preparation for the next practice, back up your orclpdb1 database, remove obsolete backups, and ensure that no failures are listed. Then exit RMAN.

SQL> **host**

$ **rman target "'/ as sysbackup'"**

…

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

…

RMAN> **delete noprompt obsolete;**

…

RMAN> **list failure;**

Database Role: PRIMARY

no failures found that match specification RMAN> **exit**

Recovery Manager complete.

$ **exit**

1. Alter your session to point to the orclpdb1 database and create a guaranteed restore point. Check your current FLASHBACK\_ON status.

Note that guaranteed restore points affect your space usage.

SQL> **alter session set container = orclpdb1;**

Session altered.

SQL> **CREATE RESTORE POINT rp1 GUARANTEE FLASHBACK DATABASE;**

Restore point created.

SQL> **SELECT FLASHBACK\_ON FROM V$DATABASE;**

FLASHBACK\_ON YES

SQL>

1. Keep the terminal windows open for the next practice.

**Practice 21-2: Performing Flashback Database**

### Overview

In this practice, you will flash back your pluggable database after some incorrect updates to the database. *This practice is for learning purposes. If you had a scenario similar to this in a production environment, you would probably choose a different solution to limit your flashback to the affected objects, rather than choose a flashback of the entire pluggable database.*

### Assumptions

You have completed practice 3-1.

You have two terminal windows open in which you are logged in as the oracle OS user and

/home/oracle/labs/DBMod\_Flashback is the current directory. Each terminal window is connected to the orclpdb1 instance.

### Tasks

1. There are several ways in which you can perform a flashback database operation. You can use a guaranteed restore point, SCN, time value, threads, and so on. This example uses the SCN, but you could also use the RP1 restore point you created in the Practice 3-1 Step 5.

Determine your current SCN. *You will need to use it in a later practice step*.

SQL> **SELECT current\_scn FROM v$database;**

CURRENT\_SCN

3285954

SQL>

1. View HR data. You will use this information for comparison during this practice.
   1. Determine the sum of the SALARY column in the HR.EMPLOYEES table.

SQL> **SELECT sum(salary) FROM hr.employees;**

SUM(SALARY)

691416

SQL>

* 1. Determine the total number of employees in department 90.

SQL> **SELECT count(\*) FROM hr.employees where department\_id=90;**

COUNT(\*)

3

SQL>

1. Execute the lab\_03\_02\_03.sql script to update tables in the HR schema. It creates issues from which you will “recover” by flashing back the database in this practice.

SQL> **@lab\_03\_02\_03.sql**

SQL> update hr.employees

1. set department\_id = 90
2. where job\_id = 'IT\_PROG' 4 /

5 rows updated.

SQL> update hr.employees e

1. set salary = least(e.salary,
2. (select (min\_salary + max\_salary)/2 \* 1.10
3. from hr.jobs j
4. where j.job\_id = e.job\_id))
5. where job\_id not like 'AD\_%' 7 /

103 rows updated.

SQL>

1. Commit your data and determine the current SCN.

SQL> **COMMIT;**

Commit complete.

SQL> **SELECT current\_scn FROM v$database;**

CURRENT\_SCN

32886103

SQL>

1. Query the data in the HR schema again and compare the results to the values you received in the queries in step 2.
   1. Determine the sum of the SALARY column in the HR.EMPLOYEES table.

SQL> **SELECT sum(salary) FROM hr.employees;**

SUM(SALARY) 679092.4

SQL>

* 1. Determine the total number of employees in department 90, and then exit SQL\*Plus.

SQL> **SELECT count(\*) FROM hr.employees where department\_id=90;**

COUNT(\*)

8

SQL> **exit**

…

$

1. You need to restore the pluggable database so that the data is as it was when you started this practice. For training purposes, use Flashback Database for this operation.
   1. Close the pluggable database orclpdb1.

$ **sqlplus / as sysdba**

…

SQL> **alter pluggable database orclpdb1 CLOSE;**

Pluggable database altered.

* 1. Use the FLASHBACK PLUGGABLE DATABASE command to flash back the database to the SCN you noted in step 1.

SQL> **flashback pluggable database orclpdb1 to scn 3285954;**

Flashback complete.

* 1. Attempt to open the pluggable database orclpdb1.

SQL> **alter pluggable database orclpdb1 open;**

alter pluggable database orclpdb1 open

\*

ERROR at line 1:

ORA-01113: file 12 needs media recovery

ORA-01110: data file 12: '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf'

* 1. Open the pluggable database orclpdb1 using the resetlogs option.

SQL> **alter pluggable database orclpdb1 open resetlogs;**

Pluggable database altered.

* 1. Switch to the ORCLPDB1 container and verify that the database was flashed back correctly by querying the HR.EMPLOYEES table again. The values should match the values you obtained in the queries in step 2.

SQL> **alter session set container = orclpdb1;**

Session altered.

SQL> **SELECT sum(salary) FROM hr.employees;**

SUM(SALARY)

691416

SQL> **SELECT count(\*) FROM hr.employees where department\_id=90;**

COUNT(\*)

3

SQL>

1. Drop the RP1 guaranteed restore point. Then exit SQL\*Plus.

SQL> **DROP RESTORE POINT rp1;**

Restore point dropped. SQL> **exit**

…

$

1. Close all terminal windows open for this practice.