

Oracle Database 19c: Backup and Recovery

Table of Contents

Course Practice Environment: Security Credentials	5
Course Practice Environment: Security Credentials	6
Practices for Lesson 1: Backup and Recovery Overview	7
Practices for Lesson 1: Overview	8
Practices for Lesson 2: Backup and Recovery Configuration	9
Practices for Lesson 2: Overview	10
Practice 2-1: Verifying that the Control File is Multiplexed	11
Practice 2-2: Configuring the Size of the Fast Recovery Area	16
Practice 2-3: Verifying that the Redo Log File Is Multiplexed	18
Practice 2-4: Configuring ARCHIVELOG Mode	25
Practices for Lesson 3: Using Recovery Manager (RMAN).....	27
Practices for Lesson 3	28
Practice 3-1: Configuring the Default Backup Destination.....	29
Practice 3-2: Setting the Date and Time Format for RMAN.....	30
Practice 3-3: Configuring RMAN Settings.....	33
Practices for Lesson 4: Backup Strategies	35
Practices for Lesson 4: Overview	36
Practices for Lesson 5: Creating Database Backups	37
Practices for Lesson 5: Overview	38
Practice 5-1: Backing Up the Control File	39
Practice 5-2: Verifying Automatic Backups of the Control File and SPFILE	44
Practice 5-3: Creating a Whole Database Backup	46
Practice 5-4: Creating Partial Database Backups.....	57
Practice 5-5: Configuring Block Change Tracking	62
Practice 5-6: Using Incremental Backups	64
Practice 5-7: Backing Up Additional Database Files	70
Practices for Lesson 6: Using Optional Backup Features	73
Practices for Lesson 6: Overview	74
Practice 6-1: Creating an Archival Backup	75
Practices for Lesson 7: Tuning RMAN Backup Performance	81
Practices for Lesson 7: Overview	82
Practice 7-1: Monitoring an RMAN Backup Job	83
Practices for Lesson 8: Recovery Catalog Overview	87
Practices for Lesson 8	88

Practices for Lesson 9: Creating a Recovery Catalog.....	89
Practices for Lesson 9: Overview	90
Practice 9-1: Creating a Recovery Catalog Owner	91
Practice 9-2: Creating the Recovery Catalog	93
Practices for Lesson 10: Managing Target Database Records	95
Practices for Lesson 10: Overview	96
Practice 10-1: Registering a Database in the Recovery Catalog	97
Practice 10-2: Configuring the Recovery Catalog for Recovery	100
Practices for Lesson 11: Using Stored Scripts	105
Practices for Lesson 11: Overview	106
Practice 11-1: Creating a Stored Script	107
Practice 11-2: Executing a Stored Script	110
Practices for Lesson 12: Creating and Using Virtual Private Catalogs	113
Practices for Lesson 12: Overview	114
Practice 12-1: Enabling the Virtual Private Database (VPD) Functionality.....	115
Practice 12-2: Creating a Virtual Private Catalog	117
Practice 12-3: Backing Up a PDB.....	119
Practices for Lesson 13: Restore and Recovery Concepts	125
Practices for Lesson 13	126
Practices for Lesson 14: Diagnosing Failures	127
Practices for Lesson 14: Overview	128
Practice 14-1: Diagnosing and Repairing Database Failure.....	129
Practices for Lesson 15: Performing Complete Recovery.....	135
Practices for Lesson 15: Overview	136
Practice 15-1: Recovering from the Loss of a System-Critical Data File.....	137
Practice 15-2: Recovering from the Loss of an Application Data File	151
Practices for Lesson 16: Performing Point-in-Time Recovery	165
Practices for Lesson 16: Overview	166
Practice 16-1: Recovering from Media Failure: Incomplete Recovery	167
Practice 16-2: Recovering a Table from a Backup	189
Practices for Lesson 17: Performing Block Media Recovery	199
Practices for Lesson 17: Overview	200
Practice 17-1: Repairing Block Corruption.....	201
Practices for Lesson 18: Performing Additional Recovery Operations.....	205
Practices for Lesson 18: Overview	206
Practice 18-1: Recovering from the Loss of a Parameter File.....	207
Practice 18-2: Restoring the Control File	210
Practice 18-3: Recovering from the Loss of All Control Files	216

Practice 18-4: Restoring the Password File.....	225
Practices for Lesson 19: Oracle Flashback Technology Overview	231
Practices for Lesson 19: Overview	232
Practices for Lesson 20: Using Logical Flashback Features	233
Practices for Lesson 20: Overview	234
Practice 20-1: Preparing to Use Flashback Technologies.....	235
Practice 20-2: Restoring a Dropped Table	238
Practice 20-3: Using Flashback Table	241
Practices for Lesson 21: Using Flashback Database	245
Practices for Lesson 21: Overview	246
Practice 21-1: Enabling Flashback Logging	247
Practice 21-2: Performing Flashback Database	249
Practices for Lesson 22: Using PDB Snapshots	253
Practices for Lesson 22	254
Practices for Lesson 23: Database Duplication Overview	255
Practices for Lesson 23	256
Practices for Lesson 24: Creating a Backup-Based Duplicate Database.....	257
Practices for Lesson 24: Overview	258
Practice 24-1: Duplicating a Database	259
Practice 24-2: Duplicating a PDB into an Existing CDB	267

**Course Practice Environment:
Security Credentials**

Course Practice Environment: Security Credentials

For operating system (OS) user names and passwords:

- If you are attending a classroom-based or live virtual class (LVC), ask your instructor or LVC producer for OS credential information.
- If you are using a self-study format, refer to the communication that you received from Oracle University for this course.

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

Product-Specific Credentials		
Product/Application	Username	Password
Database	SYS	cloud_4U
Database	SYSTEM	cloud_4U
Database	RCATOWNER	rcat_4U
Database	VPCOWNER	vpc_4U
Database (ORCLCDB and DBTEST)	SYS	cloud_4U
Database (ORCLCDB and DBTEST)	SYSTEM	cloud_4U
Database	HR	cloud_4U

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.

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 15 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
$
```

2. 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>
```

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

```
SQL> startup
ORACLE instance started.
...
Database mounted.
Database opened.
SQL>
```

- 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.

- 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>
```

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

```
SQL> CREATE PFILE FROM SPFILE;

File created.

SQL>
```

- Shut down the database instance in `IMMEDIATE` mode.

```
SQL> SHUTDOWN IMMEDIATE

Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

- Exit SQL*Plus.

```
SQL> EXIT
```

9. Create a directory for the new control file.

```
$ mkdir -p /u01/app/oracle/controlfiles_dir/ORCLCDB
$
```

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

```
$ cp $ORACLE_HOME/dbs/init.ora $ORACLE_HOME/dbs/backup_init.ora
$
```

11. 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
$
```

12. Open the PFILE (init.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/init.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'
...
$
```

13. 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>
```

14. Start the database instance.


```
SQL> STARTUP  
ORACLE instance started.  
...  
Database mounted.  
Database opened.  
SQL>
```

15. 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.

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

```
SQL> SHUTDOWN IMMEDIATE
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

- b. Exit SQL*Plus.

```
SQL> EXIT
```

- c. Use the `cp` command to re-create `control03.ctl`.

```
$ cp /u01/app/oracle/oradata/ORCLCDB/control01.ctl
/u01/app/oracle/controlfiles_dir/ORCLCDB/control03.ctl
$
```

17. Re-create the SPFILE from the updated PFILE.

- a. 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>
```

- b. Create the SPFILE.

```
SQL> CREATE SPFILE FROM PFILE;

File created.

SQL>
```

18. Start the database instance.

```
SQL> STARTUP
ORACLE instance started.

...
Database mounted.
Database opened.
SQL>
```

19. 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>
```

20. 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 5 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               string
/u01/app/oracle/fast_recovery_area
db_recovery_file_dest_size          big integer
14970M
SQL>
```

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.

2. 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.

3. 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	big integer	12G

```
SQL>
```

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 20 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;
```

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

```
SQL>
```

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.

2. Add another member to each redo log group. Name each member `redonnb.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>
```

3. 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>

```

4. Switch the log files and observe the changes.
 - a. 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;

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

SQL>

```

- b. 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>

```

- c. 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>
```

- d. 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.

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

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

SQL>
```

- e. Switch the log file.

```
SQL> ALTER SYSTEM SWITCH LOGFILE;

System altered.

SQL>
```

- f. 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.

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

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

SQL>
```

- g. Switch the log file again.

```
SQL> ALTER SYSTEM SWITCH LOGFILE;

System altered.

SQL>
```

- h. 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.

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

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

SQL>
```

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

5. To save space in your course practice environment, drop the redo log file members you created in step 4.
- a. 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>
```

- b. 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>
```

- c. 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>
```

- d. 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>
```

- e. Verify that each group now has only one member.

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

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

SQL>
```

- f. Exit from SQL*Plus.

```
SQL> exit

...
$
```

- g. Remove the physical files from the operating system.

```
$ rm /u01/app/oracle/fast_recovery_area/ORCLCDB/redo*.log
$
```

- h. Verify that the redo log files have been removed.

```
$ ls /u01/app/oracle/fast_recovery_area/ORCLCDB
archivelog  control02.ctl  onlineolog
$
```

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>
```

2. 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>
```

3. 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.
 - a. Shut down the database instance.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

- b. Start the database instance and mount the database.

```
SQL> startup mount
ORACLE instance started.

...
Database mounted.
SQL>
```

- c. Enable archiving.

```
SQL> alter database archivelog;

Database altered.

SQL>
```

- d. 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>
```

- e. Open the database.

```
SQL> alter database open;

Database altered.

SQL>
```

4. 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                string                             /u01/app/oracle/fast_recovery_area
db_recovery_file_dest_size           big integer                        12G

SQL> exit
$
```

2. 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 `~oracle/.bashrc` file:

```
export NLS_LANG=american_america.al32utf8
export NLS_DATE_FORMAT="yyyy-mm-dd:hh24:mi:ss"
```

- a. In a terminal window, invoke your editor. This example uses the `gedit` editor.

```
$ gedit ~oracle/.bashrc
```

- b. 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”.

- c. View the modified file by using the `cat` command and apply the new settings.

```
$ cat ~oracle/.bashrc
# .bashrc
...
export NLS_LANG=american_america.al32utf8
export NLS_DATE_FORMAT="yyyy-mm-dd:hh24:mi:ss"

$ . .bashrc
$
```

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

- a. 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   Type LV Size               Device Type Elapsed Time Completion
Time
-----
1        Full  1.37G          DISK          00:00:11      2019-06-
18:17:21:59
        BP Key: 1   Status: AVAILABLE Compressed: NO Tag:
TAG20190618T172148
        Piece Name:
/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      Ckp Time              Abs Fuz SCN Sparse
Name
-----
1        Full  3133091      2019-06-18:17:21:48      NO
/u01/app/oracle/oradata/ORCLCDB/system01.dbf
...
BS Key   Type LV Size               Device Type Elapsed Time Completion
Time
-----
5        Full  17.95M        DISK          00:00:01      2019-06-
18:17:22:59
        BP Key: 5   Status: AVAILABLE Compressed: NO Tag:
TAG20190618T172258

```

```
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>
```

3. **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.
 - a. Optionally, enter `SPOOL LOG TO /tmp/test.log` in your RMAN window.
 - b. Repeat the previous `LIST BACKUP` command.
 - c. Note that the output is not displayed in your window. In a second terminal window, enter `cat /tmp/test.log` to view the output.
 - d. 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 `orclpdb` 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 `SYSPBACKUP`, 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>
```

2. 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>
```

3. 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>

```

4. 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_gjl9xhwhf_.bkp comment=NONE
Finished Control File and SPFILE Autobackup at 2019-06-
18:18:04:32

```

5. Exit RMAN.

```
RMAN> exit
```

```
Recovery Manager complete.
```

```
$
```

6. 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 10 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
$
```

2. Use the `dbstart.sh` script to start the database and listener. If your database is currently running, you will see ORA-01081, which you can safely ignore.

```
$ dbstart.sh
...
SQL> ORA-01081: cannot start already-running ORACLE - shut it
down first
...
$
```

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

```
$ sqlplus / as sysdba
...
SQL>
```

4. 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>
```

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

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

Database altered.

SQL>
```

6. Exit SQL*Plus.

```
SQL> EXIT

...
$
```

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

```
$ cd /u01/app/oracle/diag/rdbms/orclcdb/orclcdb/trace
[trace]$
```

8. 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]$
```

9. 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]$
```

10. 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]$ cat orclcdb_ora_8708.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]$
```

11. 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 5 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>
```

2. 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.

3. 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>
```

2. 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				
3	850	SYS_AUX	NO	
/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf				
4	330	UNDOTBS1	YES	
/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf				
5	270	PDB\$SEED:SYSTEM	NO	
/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf				
6	330	PDB\$SEED:SYS_AUX	NO	
/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf				
7	5	USERS	NO	
/u01/app/oracle/oradata/ORCLCDB/users01.dbf				

```

8      100      PDB$SEED:UNDOTBS1      NO
/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf
9      280      ORCLPDB1:SYSTEM      YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf
10     370      ORCLPDB1:SYSAUX      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf
11     100      ORCLPDB1:UNDOTBS1      YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf
12     342      ORCLPDB1:USERS      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf
13     280      ORCLPDB2:SYSTEM      YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf
14     370      ORCLPDB2:SYSAUX      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf
15     100      ORCLPDB2:UNDOTBS1      YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf
16     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
2	36	PDB\$SEED:TEMP	32767	/u01/app/oracle/oradata/ORCLCDB/pdbseed/temp012019-05-08_01-48-43-006-AM.dbf
3	36	ORCLPDB1:TEMP	32767	/u01/app/oracle/oradata/ORCLCDB/orclpdb1/temp01.dbf
4	36	ORCLPDB2:TEMP	32767	/u01/app/oracle/oradata/ORCLCDB/orclpdb2/temp01.dbf

RMAN>

3. 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/8857B36632797E5CE0536210ED0ADAC7/backupset/2019_06_20/o1_mf_nnndf_TAG20190620T154656_gjqbmz13_.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/8857B419BF707E73E0536210ED0A54C7/backupset/2019_06_20/o1_mf_nnndf_TAG20190620T154656_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.

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

```
RMAN> LIST BACKUP;

List of Backup Sets
=====

...
BS Key   Type LV Size          Device Type Elapsed Time Completion
Time
-----
18       Full  1.40G          DISK          00:00:03      2019-06-
20:15:46:59
        BP Key: 18      Status: AVAILABLE  Compressed: NO  Tag:
TAG20190620T154656
        Piece Name:
/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      Ckp Time          Abs Fuz SCN Sparse
Name
-----
1       Full  3357562      2019-06-20:15:46:56          NO
/u01/app/oracle/oradata/ORCLCDB/system01.dbf
```


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 Type LV Size Device Type Elapsed Time Completion Time				

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 Name	LV	Type	Ckp SCN	Ckp Time	Abs Fuz SCN	Sparse	

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	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/8857888C96117610E0536210ED0A12EE/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 Name	LV	Type	Ckp SCN	Ckp Time	Abs Fuz SCN	Sparse	

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	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      Ckp time: 2019-
06-20:15:47:32

RMAN>
```

5. Exit RMAN.

```
RMAN> EXIT
```

6. 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
ol_mf_nnndf_TAG20190620T154656_gjqbmz13_.bkp

./8857B36632797E5CE0536210ED0ADAC7/backupset/2019_06_18:
total 1305840
-rw-r----- 1 oracle oinstall 66494464 Jun 18 19:39
ol_mf_nnndf_TAG20190618T193903_gj1hgq5r_.bkp
-rw-r----- 1 oracle oinstall 602128384 Jun 18 19:29
ol_mf_nnndf_TAG20190618T192926_gj1gx5j1_.bkp
-rw-r----- 1 oracle oinstall 66494464 Jun 18 18:04
ol_mf_nnndf_TAG20190618T180429_gj19xfgb_.bkp
-rw-r----- 1 oracle oinstall 602054656 Jun 18 17:22
ol_mf_nnndf_TAG20190618T172148_gj17g5kp_.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.

7. View the backup retention policy.

- a. Start RMAN and connect to the CDB root as the SYS user.

```
$ rman target /
...
connected to target database: ORCLCDB (DBID=2778750799)

RMAN>
```

- b. 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 n`, 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.

8. Manually delete obsolete files by issuing the DELETE OBSOLETE command. If there are

```
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
-----
```

obsolete files, respond `YES` when prompted. Your results will differ from this example.

```

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>

```

9. 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.

```
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.

10. 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>
```

2. 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_gjq196rc_.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>
```

3. 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.

4. Perform a partial database backup in `ORCLPDB1` directly.

- a. Start RMAN and connect to `ORCLPDB1` as the `SYS` user.

```
$ rman target SYS/<password>@orclpdb1
...
connected to target database: ORCLCDB:ORCLPDB1 (DBID=2991365572)
RMAN>
```

- b. 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>

```

5. 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.

6. Exit RMAN.

```
RMAN> EXIT
```

7. Back up the USERS tablespace in ORCLPDB2.

- a. Connect to ORCLPDB2 as the SYS user.

```
$ rman target SYS/<password>@ORCLPDB2
```

```

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

```

- b. 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>

```

- c. Exit RMAN.

```

RMAN> EXIT
...
$

```

8. Can you connect to the CDB root and perform the same operation?

- a. Start RMAN and connect to the CDB root as the `SYS` user.

```

$ rman target /
...
connected to target database: ORCLCDB (DBID=2778750799)
RMAN>

```

- b. 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.

- c. 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.
 - a. Start SQL*Plus and connect to your `ORCLCDB` database with administrator privileges.

```
$ sqlplus / as sysdba
...
SQL>
```

- b. 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	string	/u01/app/oracle/oradata/ORCLCDB

```
SQL>
```

- c. 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.

2. 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 35 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.

- a. Log in to SQL*Plus as SYSDBA.

```
$ sqlplus / as sysdba
...
SQL>
```

- b. 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
          4 ORCLPDB2          READ WRITE NO
SQL>
```

- c. Optionally, if ORCLPDB1 is MOUNTED, open the PDB and then verify the change.

```
SQL> alter pluggable database orclpdb1 open;

Pluggable database altered.

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

- d. Exit from SQL*Plus.

```
SQL> exit
...
$
```

2. 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.

3. 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.

- a. Create the new directory.

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

- b. 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.

4. Open a new terminal window and start a workload that updates your database by executing the DBMod_Backup_workload.sh script.

```

$ cd /home/oracle/labs/DBMod_Backup
$ ./DBMod_Backup_workload.sh
...
SQL>
Session altered.

SQL> Connected.
SQL> SQL> SQL> SQL> SQL> 2 3 4
2190 rows updated.

SQL>
Commit complete.
...
$

```

Note: The script updates over 2000 rows.

5. 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.

6. 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_lju4jnpc
```



```

...
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_gjxpjsb_.bkp comment=NONE
Finished Control File and SPFILE Autobackup at 2019-06-
20:21:12:49
released channel: ch1

RMAN>

```

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

```

RMAN> ALTER DATABASE DISABLE BLOCK CHANGE TRACKING;

Statement processed

RMAN>

```

8. Exit RMAN and remove your test data by executing the DBMod_Backup_cleanupINVENTORY.sh script.

```

RMAN> exit

...
$ ./DBMod_Backup_cleanupINVENTORY.sh
$

```

9. 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

```

10. 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.

11. 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>
```

12. Exit RMAN.

```
RMAN> exit
$
```

13. 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 10 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.

- a. Log in to SQL*Plus as the SYSDBA user.

```
$ sqlplus / as sysdba
...
SQL>
```

- b. 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>
```

- c. Exit from SQL*Plus.

```
SQL> exit
...
$
```

- d. 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.  
...  
$
```

2. Back up the archive log files of the `ORCLCDB` database, deleting all the archive files after the backup is complete.

- a. Log in to RMAN as `SYSBACKUP`.

```
$ rman target "/ as sysbackup"  
...  
connected to target database: ORCLCDB (DBID=2778750799)  
RMAN>
```

- b. 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/ol_mf_1_40_gjqz01js_.arc RECID=16 STAMP=1011476099
Finished backup at 2019-06-20:21:35:02
...
RMAN>
```

c. 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 20 minutes to complete this practice.

Tasks

1. Log in to RMAN.

```
$ rman target ''/ as sysbackup' "  
...  
connected to target database: ORCLCDB (DBID=2778750799)  
  
RMAN>
```

2. 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>
```

3. 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.

4. 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
```

```

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.

5. 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>
```

6. Open the database.

```
RMAN> alter database open;
```

```
Statement processed
```

```
RMAN>
```

7. 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/2019_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/2019_06_20/o1_mf_s_1011475441_gjqycl1b_.bkp RECID=39
STAMP=1011475442
```

```
Deleted 2 objects
```


RMAN>

8. 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/<password>
...
SQL>
```

2. 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
$
```

3. 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>
```

4. 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_gjt9wglh_.bkp comment=NONE
Finished Control File and SPFILE Autobackup at 2019-06-
21:18:53:02

RMAN>

```

5. Return to your first terminal session and execute the `$HOME/labs/DBMod_Backup/DBMod_Backup_MonitorBackup.sql` script approximately every 20 seconds to monitor the progress of the backup job.

```

SQL> @$HOME/labs/DBMod_Backup/DBMod_Backup_MonitorBackup

```

SID	SERIAL#	CONTEXT	SO FAR	TOTALWORK	%_COMPLETE
267	40731	5	0	638720	0

```

SQL> /

```

SID	SERIAL#	CONTEXT	SO FAR	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.

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

```

RMAN> exit
...
$

```


7. Return to your SQL*Plus session. Log out of SQL*Plus.

```
SQL> exit  
...  
$
```

8. 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 `RCATCDB` database instance.

Timing Estimate

It should take approximately 10 minutes to complete this practice.

Tasks

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

```
$ . oraenv
ORACLE_SID = [orclcdb] ? rcatcdb
The Oracle base remains unchanged with value /u01/app/oracle
$
```

2. Start the `RCATCDB` database.
 - a. Log in to SQL*Plus as `SYSDBA`.

```
$ sqlplus / as sysdba
...
Connected to an idle instance.

SQL>
```

- b. Start the instance.

```
SQL> startup
ORACLE instance started.
...
Database opened.
SQL>
```

- c. Exit from SQL*Plus.

```
SQL> exit
...
$
```

3. Create a tablespace named `RCATTBS` for the recovery catalog.
 - a. Use SQL*Plus to log in to the `RCATPDB` PDB as the `SYS` user with the `SYSDBA` role. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

```
$ sqlplus sys@rcatpdb as sysdba
...
Enter password: *****
...
SQL>
```

- b. Verify that the RCATPDB PDB is open. If it is not, open the RCATPDB PDB using the ALTER DATABASE OPEN command.

```
SQL> show pdbs

          CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          3 RCATPDB                                READ WRITE NO

SQL>
```

- c. Create the RCATTBS tablespace with a single 15 MB datafile.

```
SQL> CREATE TABLESPACE rcattbs DATAFILE
'/u01/app/oracle/oradata/RCATCDB/rcatpdb/rcat01.dbf' SIZE 15M
REUSE;

Tablespace created.

SQL>
```

4. 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 `<password>`.

```
SQL> CREATE USER rcatowner IDENTIFIED BY <password> DEFAULT
TABLESPACE rcattbs QUOTA unlimited on rcattbs;

User created.

SQL>
```

5. 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 `RCATCDB` database instance.

Timing Estimate

It should take approximately 5 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@rcatpdb
...
recovery catalog database Password: *****
connected to recovery catalog database

RMAN>
```

2. 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 `ORCLCDB` 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 `ORCLCDB` database in the recovery catalog that you created in a previous practice.

Assumptions

A terminal window is open.

Timing Estimate

It should take approximately 5 minutes to complete this practice.

Tasks

1. Set the environment variables for the `ORCLCDB` database.

```
$ . oraenv
ORACLE_SID = [rcatcdb] ? orclcdb
The Oracle base remains unchanged with value /u01/app/oracle
$
```

2. 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@rcatpdb
...
connected to target database: ORCLCDB (DBID=2778750799)
recovery catalog database Password: *****
connected to recovery catalog database

RMAN>
```

3. Register the database in the catalog.

```
RMAN> register database;

database registered in recovery catalog
starting full resync of recovery catalog
full resync complete

RMAN>
```

4. To verify that the registration was successful, execute the `REPORT SCHEMA` command. The output should list all of the data files associated with the `ORCLCDB` database that have been registered in the recovery catalog.

```

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
4      330      UNDOTBS1                YES
/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf
5      270      PDB$SEED:SYSTEM        NO
/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf
6      330      PDB$SEED:SYSAUX        NO
/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf
7       5      USERS                  NO
/u01/app/oracle/oradata/ORCLCDB/users01.dbf
8      100      PDB$SEED:UNDOTBS1      NO
/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf
9      280      ORCLPDB1:SYSTEM        YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf
10     370      ORCLPDB1:SYSAUX        NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf
11     100      ORCLPDB1:UNDOTBS1      YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf
12     342      ORCLPDB1:USERS         NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf
13     280      ORCLPDB2:SYSTEM        YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf
14     370      ORCLPDB2:SYSAUX        NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf
15     100      ORCLPDB2:UNDOTBS1      YES
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf
16     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
2	36	PDB\$SEED:TEMP	32767	/u01/app/oracle/oradata/ORCLCDB/pdbseed/temp012019-05-08_01-48-43-006-AM.dbf
3	36	ORCLPDB1:TEMP	32767	/u01/app/oracle/oradata/ORCLCDB/orclpdb1/temp01.dbf
4	36	ORCLPDB2:TEMP	32767	/u01/app/oracle/oradata/ORCLCDB/orclpdb2/temp01.dbf

RMAN>

5. 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 `RCATCDB`, enable archive log mode, and back up your `RCATCDB` database.

You back up the recovery catalog to implement a backup strategy of incremental backups applied to image copies. This provides a method of fast restore by switching to the image copy rather than copying the backups back to the original location.

Assumptions

A terminal window is open.

Timing Estimate

It should take approximately 25 minutes to complete this practice.

Tasks

1. Configure the retention policy with at least redundancy 2.
 - a. Set your environment variables to point to the `RCATCDB` database.

```
$ . oraenv
ORACLE_SID = [orclpdb] ? rcatcdb
The Oracle base remains unchanged with value /u01/app/oracle
$
```

- b. Start RMAN and use the recovery catalog database (`RCATCDB`) 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: *****
connected to target database: RCATCDB (DBID=2073894879)

RMAN>
```

- c. 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
RCATCDB 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>

```

- d. Exit RMAN.

```

RMAN> exit

```

```

...

```

```

$

```

2. To configure the fast recovery area for RCATCDB, perform the following:

- a. Ensure that the environment variables are set for the RCATCDB database.

```

$ . oraenv

```

```

ORACLE_SID = [rcatcdb] ? rcatcdb

```

```

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

```

```

$

```

- b. Log in to SQL*Plus.

```

$ sqlplus / as sysdba

```

```

...

```

```

SQL>

```

- c. Set the size of the fast recovery area for RCATCDB to 12 GB.

```

SQL> ALTER SYSTEM SET db_recovery_file_dest_size=12G SCOPE=BOTH;

```

```

System altered.

```

```

SQL>

```

3. Verify that the recovery catalog database is in ARCHIVELOG mode.

- a. Check whether RCATCDB 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>

```

- b. 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>
```

- c. Put the database in ARCHIVELOG mode and then open the database.

```
SQL> ALTER DATABASE ARCHIVELOG;

Database altered.

SQL> ALTER DATABASE OPEN;

Database altered.

SQL>
```

- d. 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>
```

- e. Verify that RCATPDB is open.

```
SQL> show pdbs

  CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
      2 PDB$SEED                                READ ONLY  NO
      3 RCATPDB                                READ WRITE NO
```

- f. Optionally, if RCATPDB is not open, open the PDB and save the state so that the RCATPDB PDB is opened every time the CDB is opened.

```
SQL> ALTER PLUGGABLE DATABASE rcatpdb open;

Pluggable database altered.

SQL> ALTER PLUGGABLE DATABASE rcatpdb save state;

Pluggable database altered.

SQL>
```

- g. Exit from SQL*Plus.

```
SQL> exit

...

$
```

4. Back up the recovery catalog database as an image copy and a base for incremental backups. This provides a method of fast restore by switching to the image copy rather than copying the backups back to the original location. Use a terminal window where your environment variables point to the RCATCDB database.

- a. 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: *****
connected to target database: RCATCDB (DBID=2073894879)

RMAN>
```

- b. Back up the recovery catalog database as an image copy and a base for incremental backups. Review the output and find the data file for the tablespace that you created for the recovery catalog.

```
RMAN> BACKUP AS COPY INCREMENTAL LEVEL 0 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/RCATCDB/system01.dbf
output file
name=/u01/app/oracle/fast_recovery_area/RCATCDB/datafile/ol_mf_s
```

```

system_gkdy3bqy_.dbf tag=TAG20190628T202826 RECID=4
STAMP=1012163316
channel ORA_DISK_1: datafile copy complete, elapsed time:
00:00:15
channel ORA_DISK_1: starting datafile copy
...
input datafile file number=00013
name=/u01/app/oracle/oradata/RCATCDB/rcatpdb/rcat01.dbf
output file
name=/u01/app/oracle/fast_recovery_area/RCATCDB/8C6740C0D1762DE1
E0532811ED0A894F/datafile/ol_mf_rcattbs_gkdy5btc_.dbf
tag=TAG20190628T202826 RECID=13 STAMP=1012163371
channel ORA_DISK_1: datafile copy complete, elapsed time:
00:00:01
channel ORA_DISK_1: starting datafile copy
...
Starting Control File and SPFILE Autobackup at 2019-06-
28:20:29:34
piece
handle=/u01/app/oracle/fast_recovery_area/RCATCDB/autobackup/201
9_06_28/ol_mf_s_1012163374_gkdy5gkc_.bkp comment=NONE
Finished Control File and SPFILE Autobackup at 2019-06-
28:20:29:35

RMAN>

```

c. Exit RMAN.

```

RMAN> exit
...
$

```

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, ORCLCDB.

```
$ . oraenv
ORACLE_SID = [rcatcdb] ? orclcdb
The Oracle base remains unchanged with value /u01/app/oracle
$
```

2. Invoke RMAN and connect to the target database and the RCATCDB recovery catalog. Refer to the *Course Practice Environment: Security Credentials* document for the correct password for the recovery catalog owner.

```
$ rman target / catalog rcatowner@rcatpdb
...
connected to target database: ORCLCDB (DBID=2778750799)
recovery catalog database Password: *****
connected to recovery catalog database

RMAN>
```

3. Display the names of the stored scripts.

```
RMAN> LIST SCRIPT NAMES;
List of Stored Scripts in Recovery Catalog

          No scripts in recovery catalog

RMAN>
```

4. Create a local script named `db_arch_bkup` to perform a database backup, including the archived log files.

```
RMAN> CREATE SCRIPT db_arch_bkup
2> {
3> BACKUP DATABASE PLUS ARCHIVELOG;
4> }

created script db_arch_bkup

RMAN>
```

5. 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>
```

6. Modify the `db_arch_bu` script so that it also deletes obsolete archived log files.

```
RMAN> REPLACE SCRIPT db_arch_bkup
2> {
3> BACKUP DATABASE PLUS ARCHIVELOG;
4> DELETE OBSOLETE;
5> }

replaced script db_arch_bkup

RMAN>
```

7. 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>
```

8. 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 10 minutes to complete this practice.

Tasks

1. Set the environment variables for the target database, ORCLCDB.

```
$ . oraenv
ORACLE_SID = [orclcdb] ?
The Oracle base remains unchanged with value /u01/app/oracle
$
```

2. Invoke RMAN and connect to the target database and the `RCATCDB` recovery catalog. Refer to the *Course Practice Environment: Security Credentials* document for the correct password for the recovery catalog owner.

```
$ rman target / catalog rcatowner@rcatpdb
...
connected to target database: ORCLCDB (DBID=2778750799)
recovery catalog database Password:
connected to recovery catalog database

RMAN>
```

3. Execute the `db_arch_bkup` script. Be sure to respond YES when prompted to delete the obsolete backups.

```
RMAN> run
2> {
3> EXECUTE SCRIPT db_arch_bkup;
4> }

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/ol_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/
ol_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/ol_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/ol_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/ol_mf_nnndf_TAG20190628T
211946_gkf15cs1_.bkp RECID=72 STAMP=1012166443
Deleted 13 objects

RMAN>

```

4. Verify that your script created a full database backup.

```

RMAN> list backup;

List of Backup Sets
=====

```

```

BS Key   Type LV Size          Device Type Elapsed Time Completion
Time
-----
299      Full   1.42G          DISK          00:00:03    2019-06-
21:15:39:37
          BP Key: 319    Status: AVAILABLE  Compressed: NO  Tag:
KEEP_DB_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      Low SCN      Low Time          Next SCN      Next
Time
-----
   1    43      3504011    2019-06-28:21:19:40 3504086      2019-
06-28:21:20:50

BS Key   Type LV Size          Device Type Elapsed Time Completion
Time
-----
476      Full   17.95M          DISK          00:00:01    2019-06-
28:21:20:54
          BP Key: 478    Status: AVAILABLE  Compressed: NO  Tag:
TAG20190628T212053
          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>

```

5. 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 of `ORCLPDB1`.

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 15 minutes to complete this practice.

Tasks

1. Set the environment variables for the recovery catalog database (RCATCDB).

```
$ . oraenv
ORACLE_SID = [orclpdb] ? rcatcdb
The Oracle base remains unchanged with value /u01/app/oracle
$
```

2. 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.

```
$ sqlplus sys@rcatpdb as sysdba
...
Enter password: *****
...
SQL>
```

3. 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 ...

$
```

4. Upgrade the recovery catalog.

- a. 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@rcatpdb
...
recovery catalog database Password:
connected to recovery catalog database

RMAN>
```

- b. 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>
```

- c. Exit from RMAN.

```
RMAN> exit
...
$
```

Practice 12-2: Creating a Virtual Private Catalog

Overview

In this practice, you create a new user and create a virtual private catalog.

Assumptions

Timing Estimate

It should take approximately 15 minutes to complete this practice.

Tasks

1. Invoke SQL*Plus and connect to the recovery catalog database with administrator privileges. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

```
$ sqlplus sys@rcatpdb as sysdba
...
Enter password: *****
...
SQL>
```

2. Create a tablespace for virtual private catalog users.

```
SQL> CREATE TABLESPACE vpctbs DATAFILE
'/u01/app/oracle/oradata/RCATCDB/rcatpdb/vpc01.dbf' SIZE 5M
REUSE;

Tablespace created.

SQL>
```

3. Create a new user who will own the virtual private catalog. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

```
SQL> CREATE USER vpcowner IDENTIFIED BY <password>
2 DEFAULT TABLESPACE vpctbs
3 QUOTA UNLIMITED ON vpctbs;

User created.

SQL>
```

4. Grant the `CREATE SESSION` privilege to the new user who owns the virtual private catalog.

```
SQL> GRANT CREATE SESSION TO vpcowner;  
  
Grant succeeded.  
  
SQL>
```

5. Exit SQL*Plus.

```
SQL> exit  
...  
$
```

6. Invoke RMAN and connect to the recovery catalog database as the base recovery catalog owner. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

```
$ rman catalog rcatowner@rcatpdb  
...  
recovery catalog database Password: *****  
connected to recovery catalog database  
  
RMAN>
```

7. Grant access to the metadata for the `ORCLPDB1` PDB to the virtual private catalog owner.

```
RMAN> GRANT CATALOG FOR PLUGGABLE DATABASE orclpdb1 TO vpcowner;  
  
Grant succeeded.  
  
RMAN>
```

8. Exit RMAN.

```
RMAN> exit  
...  
$
```

Practice 12-3: Backing Up a PDB

Overview

In this practice, you will connect to the virtual private catalog and back up the ORCLPDB1 PDB.

Assumptions

You have an open terminal window.

Timing Estimate

It should take approximately 15 minutes to complete this practice.

Tasks

1. Invoke RMAN and connect to the recovery catalog database as the virtual private catalog owner (not the base recovery catalog owner). Connect as TARGET to ORCLPDB1.
 - a. Use `oraenv` to set the environment variables for the ORCLCDB database.

```
$ . oraenv
ORACLE_SID = [rcatcdb] ? orclcdb
The Oracle base remains unchanged with value /u01/app/oracle
$
```

- b. Invoke RMAN and connect to ORCLPDB1. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

```
$ rman target sys@orclpdb1
...
target database Password:
connected to target database: ORCLCDB:ORCLPDB1 (DBID=2991365572)

RMAN>
```

- c. Connect to recovery catalog database, RCATPDB, as the VPCOWNER user. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

```
RMAN> CONNECT CATALOG vpcowner@rcatpdb;

recovery catalog database Password:
connected to recovery catalog database

RMAN>
```

2. Perform a backup of the ORCLPDB1 PDB.

```
RMAN> backup pluggable database orclpdb1;

Starting backup at 2019-07-01:18:31:40
allocated channel: ORA_DISK_1
```

```

channel ORA_DISK_1: SID=38 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 2019-07-01:18:31:43
channel ORA_DISK_1: finished piece 1 at 2019-07-01:18:31:58
piece
handle=/u01/app/oracle/fast_recovery_area/ORCLCDB/8857B36632797E
5CE0536210ED0ADAC7/backupset/2019_07_01/o1_mf_nnndf_TAG20190701T
183142_gknndh1r_.bkp tag=TAG20190701T183142 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:15
Finished backup at 2019-07-01:18:31:58

RMAN>

```

3. List the backups in the virtual private catalog.

```

RMAN> list backup;

List of Backup Sets
=====

BS Key    Type LV Size       Device Type Elapsed Time Completion
Time
-----
300       Full  575.46M    DISK          00:00:01    2019-06-
21:15:39:50

        BP Key: 320    Status: AVAILABLE   Compressed: NO   Tag:
KEEP_DB_TAG

        Piece Name: /home/oracle/backup/2uu4lp65_1_1
        Keep: BACKUP_LOGS          Until: 2020-06-20:15:39:34

List of Datafiles in backup set 300
File LV Type Ckp SCN      Ckp Time              Abs Fuz SCN Sparse
Name
-----
9       Full 3470037    2019-06-21:14:57:43          NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf
10      Full 3470037    2019-06-21:14:57:43          NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf

```

```

11      Full 3470037      2019-06-21:14:57:43      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf
12      Full 3470037      2019-06-21:14:57:43      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

BS Key   Type LV Size          Device Type Elapsed Time Completion
Time
-----
434      Full      575.41M      DISK          00:00:06      2019-06-
28:21:20:19
          BP Key: 440 Status: AVAILABLE Compressed: NO Tag:
TAG20190628T211946
          Piece Name:
/u01/app/oracle/fast_recovery_area/ORCLCDB/8857B36632797E5CE0536
210ED0ADAC7/backupset/2019_06_28/o1_mf_nnndf_TAG20190628T211946_
gkf14fd8_.bkp
...
List of Datafiles in backup set 510
File LV Type Ckp SCN      Ckp Time          Abs Fuz SCN Sparse
Name
-----
9      Full 3515990      2019-07-01:18:31:43      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf
10      Full 3515990      2019-07-01:18:31:43      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf
11      Full 3515990      2019-07-01:18:31:43      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf
12      Full 3515990      2019-07-01:18:31:43      NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

RMAN>

```

4. Exit RMAN.

```

RMAN> exit
...
$

```

5. Now connect as the recovery catalog owner and list the backups.

- a. Connect to RMAN. Refer to the *Course Practice Environment: Security Credentials* document for the correct password.

```

$ rman target '/' as sysbackup' catalog rcatowner@rcatpdb
...
connected to target database: ORCLCDB (DBID=2778750799)
recovery catalog database Password:
connected to recovery catalog database

```

```
RMAN>
```

- b. List the backups in the recovery catalog.

```
RMAN> list backup;
```

```
starting full resync of recovery catalog
full resync complete
```

```
List of Backup Sets
```

```
=====
```

BS Key	Type	LV	Size	Device	Type	Elapsed Time	Completion Time
--------	------	----	------	--------	------	--------------	-----------------

299	Full		1.42G	DISK		00:00:03	2019-06-21:15:39:37
-----	------	--	-------	------	--	----------	---------------------

```
BP Key: 319    Status: AVAILABLE    Compressed: NO    Tag:
KEEP_DB_TAG
```

```
Piece Name: /home/oracle/backup/2tu4lp5m_1_1
```

```
Keep: BACKUP_LOGS    Until: 2020 06-20:15:39:34
```

```
List of Datafiles in backup set 299
```

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

1		Full	3470197		2019-06-21:14:57:45				NO
/u01/app/oracle/oradata/ORCLCDB/system01.dbf									
3		Full	3470197		2019-06-21:14:57:45				NO
/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf									
4		Full	3470197		2019-06-21:14:57:45				NO
/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf									
7		Full	3470197		2019-06-21:14:57:45				NO
/u01/app/oracle/oradata/ORCLCDB/users01.dbf									

BS Key	Type	LV	Size	Device	Type	Elapsed Time	Completion Time
--------	------	----	------	--------	------	--------------	-----------------

300	Full		575.46M	DISK		00:00:01	2019-06-21:15:39:50
-----	------	--	---------	------	--	----------	---------------------

```
BP Key: 320    Status: AVAILABLE    Compressed: NO    Tag:
KEEP_DB_TAG
```

```
Piece Name: /home/oracle/backup/2uu4lp65_1_1
```

```
Keep: BACKUP_LOGS    Until: 2020 06-20:15:39:34
```

```
List of Datafiles in backup set 300
```

```
Container ID: 3, PDB Name: ORCLPDB1
```


File Name	LV	Type	Ckp SCN	Ckp Time	Abs Fuz SCN	Sparse
9		Full	3470037	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf						
10		Full	3470037	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf						
11		Full	3470037	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf						
12		Full	3470037	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf						

BS Key	Type	LV	Size	Device	Type	Elapsed Time Completion
Time						

301	Full		574.20M	DISK		00:00:01 2019-06-21:15:39:58
BP Key: 321 Status: AVAILABLE Compressed: NO Tag: KEEP_DB_TAG						
Piece Name: /home/oracle/backup/2vu4lp6d_1_1						
Keep: BACKUP_LOGS Until: 2020 06-20:15:39:34						
List of Datafiles in backup set 301						
Container ID: 4, PDB Name: ORCLPDB2						
File Name	LV	Type	Ckp SCN	Ckp Time	Abs Fuz SCN	Sparse

13		Full	3470036	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf						
14		Full	3470036	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf						
15		Full	3470036	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf						
16		Full	3470036	2019-06-21:14:57:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf						
...						
List of Datafiles in backup set 510						
Container ID: 3, PDB Name: ORCLPDB1						
File Name	LV	Type	Ckp SCN	Ckp Time	Abs Fuz SCN	Sparse

9		Full	3515990	2019-07-01:18:31:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf						
10		Full	3515990	2019-07-01:18:31:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf						
11		Full	3515990	2019-07-01:18:31:43		NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf						

```
12          Full 3515990      2019-07-01:18:31:43          NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf

RMAN>
```

c. Exit RMAN.

```
RMAN> exit
...
$
```

6. Close the terminal window opened for this practice.

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/onlineolog/redo3a.log  
' TO GROUP 3  
$
```

2. 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  
$
```

3. 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  
$
```

4. 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  
-----  
          2 PDB$SEED                                READ ONLY NO  
          3 ORCLPDB1                                MOUNTED  
          4 ORCLPDB2                                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
$

```

5. 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>

```

6. 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
2. Automatic repairs may be available if you shutdown the
database and restart it in mount mode
3. If you restored the wrong version of data file
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf, then
replace it with the correct one
4. If you restored the wrong version of data file
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf, then
replace it with the correct one
5. If you restored the wrong version of data file
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf, then
replace it with the correct one
6. 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>
```

7. 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/ol_mf_bar91tbs_gkpxfylo_.dbf
destination for restore of datafile 00190:
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bar91tbs01.dbf
```

```
...
```

```
repair failure complete
```

```
RMAN>
```

8. Open the ORCLPDB1 pluggable database.

```
RMAN> alter pluggable database orclpdb1 open;

Statement processed

RMAN> exit
...
$
```

9. Confirm that ORCLPDB1 is open.

```
$ sqlplus / as sysdba
...
SQL> show PDBS

      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          2 PDB$SEED                                READ ONLY  NO
          3 ORCLPDB1                                READ WRITE NO
          4 ORCLPDB2                                MOUNTED

SQL> exit
$
```

10. 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 `/tmp/cleanup.log` file.

```
$ ./cleanup_02_01.sh
$
```

11. 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. If you need to restart the database and listener, use the `dbstart.sh` script.

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/initorcldb.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  
$
```

2. Perform a cold database backup.

- a. Create a directory to preserve the backup.

```
$ mkdir -p /u01/app/oracle/donttouch
```

- b. Start RMAN.

```
$ rman target /  
...  
connected to target database: ORCLCDB (DBID=2778750799)  
RMAN>
```

- c. Shut down the database, and bring it to the mount stage.

```
RMAN> SHUTDOWN IMMEDIATE;  
...  
RMAN> STARTUP MOUNT;  
...
```

- d. 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";  
...
```

- e. Open the database.

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

- f. Exit RMAN.

```
RMAN> EXIT
```

3. Create a hot backup to the RMAN configured location.

- a. 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
```

- b. 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

```

c. 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

$
```

3. Attempt an administrative task, such as creating a user.
 - a. 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/<password>

...
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

$

```

4. Consider your recovery options.
 - a. **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).
 - b. **Question:** Which methods can you use to recover the data file?
 - c. **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.
5. 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.
 - a. If the SQL*Plus login displays "ORA-01034: ORACLE not available", then the database instance is down. **Action:** Read case 3 in step 6.
 - b. 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
 - c. If the `rman target /` command shows an error and immediately disconnects then the database instance is down.
 - d. 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`.
6. The exact commands that you will use vary based on the state of the database instance. Follow each case in order.
 - a. 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

```

- b. 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
```

- c. 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>
```

2. 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>
```

3. Open the CDB root.

```
RMAN> ALTER DATABASE OPEN;

Statement processed

RMAN>
```

4. Open all PDBs.

```
RMAN> ALTER PLUGGABLE DATABASE ALL OPEN;

Statement processed

RMAN>
```

5. Exit RMAN.

```
RMAN> EXIT

Recovery Manager complete.
$
```

6. Start SQL*Plus and connect to the CDB root as the SYSDBA user.

```
$ sqlplus / as sysdba

...

SQL>
```


7. Show pdb status.

```
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

8. 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.

```
$ $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
$
```

2. **Window 1:** Try to connect to the database.

```
$ sqlplus / as sysdba
```

```
...
```

3. If the database appears to be running attempt a simple query.

```
SQL> select username from dba_users;
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
```

4. **Window 1:** Shut down the database instance.

```
SQL> shutdown abort
...
```

5. **Window 1:** Exit SQL*Plus.

```
SQL> EXIT
...
$
```

6. **Window 1:** Start RMAN and connect to the target database.

```
$ rman target /
...
connected to target database (not started)

RMAN>
```

7. **Window 1:** Start the database instance in MOUNT mode.

```
RMAN> STARTUP MOUNT;
database mounted
...

RMAN>
```

8. **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>
```

9. **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>
```

10. **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>
```

11. **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_gk9vtll_.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_gk9vtmm_.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_gk9vtll_.arc thread=1 sequence=118
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
06_27/o1_mf_1_119_gk9vtmm_.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>
```

12. **Window 1:** Open all the PDBs.

```
RMAN> ALTER PLUGGABLE DATABASE ALL OPEN;

Statement processed

RMAN>
```

13. **Window 1:** Exit RMAN.

```
RMAN> EXIT

Recovery Manager complete.
$
```

14. **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.

`orclpdb` 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.

$
```

2. Start RMAN and connect to the CDB root as the `SYS` user.

```
$ rman target /
...
connected to target database: ORCLCDB (DBID=1500451933)

RMAN>
```

3. 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>

```

4. 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.

$
```

2. Create an application table and insert data into it.

- a. Connect to the CDB root as the SYS user with the SYSDBA privilege.

```
$ sqlplus / AS SYSDBA
...
SQL>
```

- b. Connect to ORCLPDB1 as the SYS user.

```
SQL> ALTER SESSION SET CONTAINER = ORCLPDB1;

Session altered.

SQL>
```

- c. 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
```

- d. 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.

```
SQL> SHOW PDBs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
3	ORCLPDB1	READ WRITE	NO

- e. Exit and reconnect to the database as sys.

```
SQL> Exit
...
$ sqlplus / as sysdba
...
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

- f. If PDB ORCLPDB is open, close it. This may show an error.

```
SQL> ALTER PLUGGABLE DATABASE ORCLPDB1 CLOSE;

Pluggable database altered.
```

- g. 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'
```

3. 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).

2. **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.

3. Start RMAN and connect to ORCLPDB1 as the SYS user. Refer to *Course Practice Environment: Security Credentials* for the password value.

```
$ rman target SYS/<password>@ORCLPDB1
...
connected to target database: ORCLCDB:ORCLPDB1 (DBID=135052775,
not open)
RMAN>
```

4. Issue the `REPORT SCHEMA` command to list the names of the data files (permanent and temporary) and tablespaces for ORCLPDB1. In this example, data file number 177 is part of the `TBS_APP` tablespace in ORCLPDB1 and shows a size of 0MB. Identify the entry for the `TBS_APP` tablespace on your system.

```
RMAN> REPORT SCHEMA;

using target database control file instead of recovery catalog
Report of database schema for database with db_unique_name
ORCLCDB

List of Permanent Datafiles
=====
File Size(MB) Tablespace          RB segs Datafile Name
-----
160  290      SYSTEM              NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf
161  400      SYSAUX              NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf
```

```

162 105      UNDOTBS1          NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf
163 360      USERS            NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf
172 7        INVENTORY        NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/INVENTORY01.DBF
177 0      TBS_APP            NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/tbs_app01.dbf

```

List of Temporary Files

=====

File	Size (MB)	Tablespace	Maxsize (MB)	Tempfile Name
3	36	TEMP	32767	/u01/app/oracle/oradata/ORCLCDB/orclpdb1/temp01.dbf

RMAN>

- Optionally, if ORCLPDB1 were open, you would close ORCLPDB1 in IMMEDIATE mode (which puts the PDB into MOUNTED mode) before restoring the PDB; otherwise, you will get the error "cannot obtain exclusive enqueue for datafile...".

```
RMAN> SHUTDOWN IMMEDIATE;
```

- Restore ORCLPDB1.

```
RMAN> RESTORE DATABASE;
```

Starting restore at 27-JUN-19

allocated channel: ORA_DISK_1

channel ORA_DISK_1: SID=274 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 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
```

```
RMAN>
```

7. Recover the database. Notice the line "starting media recovery." If you had tried to restore and recover just the data file or just the tablespace, you might have encountered media recovery errors for other data files in the tablespace.

```
RMAN> RECOVER DATABASE;
```

```
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 127 is already on disk
as file
```

```
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_06_27
/o1_mf_1_127_gkb2lqtd_.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
```

```
media recovery complete, elapsed time: 00:00:00
```

```
Finished recover at 27-JUN-19
```

```
RMAN>
```

8. Start ORCLPDB1.

```
RMAN> STARTUP
```

```
database opened
```

```
RMAN>
```

9. Issue the `REPORT SCHEMA` command again. Notice that the `TBS_APP` tablespace now has a size of 800MB.

```
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
-----
160  290      SYSTEM              NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/system01.dbf
...
177  800      TBS_APP              NO
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/tbs_app01.dbf

List of Temporary Files
=====
File Size(MB) Tablespace          Maxsize(MB) Tempfile Name
-----
3    36      TEMP                  32767
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/temp01.dbf
RMAN>
```

10. Exit RMAN.

```
RMAN> EXIT

Recovery Manager complete.
$
```

11. Try again to create a table named `ODR.TEST` in `ORCLPDB1`.

- a. Connect to the CDB root as the `SYS` user with the `SYSDBA` privilege.

```
$ sqlplus / AS SYSDBA
...
SQL>
```

- b. Connect to `ORCLPDB1` as the `SYS` user.

```
SQL> ALTER SESSION SET CONTAINER = ORCLPDB1;

Session altered.

SQL>
```

- c. 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>
```

- d. 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.

12. Exit SQL*Plus.

```
SQL> EXIT  
...  
$
```

Use the REPAIR Command

In this section, you re-create the environment where you have a missing data file. You then use RMAN's `REPAIR` command to perform all the necessary operations (for example, restore and recovery) to fully recover the data file.

1. Execute the script to remove the data file named `tbs_app01.dbf` from `ORCLPDB1` again.

```
$ $HOME/labs/DBMod_Recovery/RMAN_crash_app.sh  
  
System altered.  
  
System altered.  
  
$
```

2. Create an application table and try to insert data into it.

- a. Connect to the CDB root as the `SYS` user with the `SYSDBA` privilege.

```
$ sqlplus / AS SYSDBA  
...  
SQL>
```

- b. Connect to ORCLPDB1 as the SYS user.

```
SQL> ALTER SESSION SET CONTAINER = ORCLPDB1;

Session altered.

SQL>
```

- c. Show the open mode of ORCLPDB1. The result shows that ORCLPDB1 is still open.

```
SQL> SHOW PDBS

      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          4 ORCLPDB1                                READ WRITE NO

SQL>
```

- d. Create a table named ODR.TEST. and insert a row into the 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> 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
```

- e. Show the open mode of ORCLPDB1. The result may show that ORCLPDB1 is still open or it may fail.

```
SQL> SHOW PDBS

      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          3 ORCLPDB1                                READ WRITE NO
```


- f. Exit and reconnect to the database as sys.

```
SQL> Exit
...
$ sqlplus / as sysdba
...
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

- g. If ORCLPDB1 is still open, close it.

```
SQL> ALTER PLUGGABLE DATABASE ORCLPDB1 CLOSE;
```

Pluggable database altered.

- h. 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'
```

3. Exit SQL*Plus.

```
SQL> EXIT
...
$
```

4. Start RMAN and log in to the target database as the SYS user.

```
$ rman target /
...
connected to target database: ORCLCDB (DBID=2778750799)

RMAN>
```

5. 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>

```

6. Open ORCLPDB1.

```

RMAN> ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN;

Statement processed

RMAN>

```

7. Exit RMAN.

```

RMAN> EXIT

Recovery Manager complete.
$

```

8. Try to insert data into the ODR.TEST table again.

- a. Start SQL*Plus and connect to ORCLPDB1 as the SYSTEM user. Refer to *Course Practice Environment: Security Credentials* for the password value.

```

$ sqlplus SYSTEM/<password>@ORCLPDB1
...
SQL>

```

- b. 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>

```

- c. Commit the transaction.

```
SQL> commit;  
  
Commit complete.  
  
SQL>
```

- d. Exit SQL*Plus.

```
SQL> EXIT  
  
...  
$
```

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

Practices for Lesson 16: Overview

Overview

In these practices, you will perform a point-in-time (also known as incomplete) recovery of the database. You will also recover a table from a backup set without affecting other objects in the tablespace or schema.

Practice 16-1: Recovering from Media Failure: Incomplete Recovery

Overview

In this practice, you set up a scenario that requires an incomplete recovery. Then you perform the steps that are needed when an archive log is missing after the last backup (and transactions exist that cannot be re-created); therefore, complete recovery is not possible.

Assumptions

A full backup exists and the archive log files from the time of the backup to the current time are available.

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.

You have a recovery catalog instance, `rcatpdb`, available on your machine.

Tasks

1. Verify that the `rcatcdb` and `rcatpdb` are open.
 - a. Set the environment for the `rcatcdb` database.

```
$ . oraenv
ORACLE_SID = [orclcdb] ? rcatcdb
The Oracle base remains unchanged with value /u01/app/oracle
```

- b. Verify that the `rcatpdb` is open, then exit SQL*Plus.

```
$ sqlplus / as sysdba
...
SQL> SHOW PDBS
      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          2 PDB$SEED                                READ ONLY  NO
          3 RCATPDB                                  READ WRITE NO

SQL> EXIT
```

2. Synchronize the catalog with the database.
 - a. Set the environment for the `orclcdb` database instance

```
$ . oraenv
ORACLE_SID = [rcatcdb] ? orclcdb
The Oracle base remains unchanged with value /u01/app/oracle
```

- b. Connect to the target (database, `orclpdb`) and the recovery catalog (`rcatpdb`) and resynchronize the catalog with the database control file. Refer to the *Course Practice Environment: Security Credentials* for the passwords.

```
$ rman target / catalog rcatowner/<password>@rcatpdb
...
connected to target database: ORCLCDB (DBID=2778750799)
connected to recovery catalog database

RMAN> RESYNC CATALOG;

starting full resync of recovery catalog
full resync complete

RMAN> Exit
```

3. Start RMAN and connect to the ORCLCDB root as the SYS user.

```
$ rman target /
...
connected to target database: ORCLCDB (DBID= 2778750799)

RMAN>
```

4. Back up ORCLPDB1.

Note: A database backup should be taken after every recovery.

```
RMAN> BACKUP PLUGGABLE DATABASE ORCLPDB1;

Starting backup at 03-JUL-19
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=272 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=00197
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=00198
name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf
```

```

input datafile file number=00172
name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/INVENTORY01.DBF
channel ORA_DISK_1: starting piece 1 at 03-JUL-19
channel ORA_DISK_1: finished piece 1 at 03-JUL-19
piece
handle=/u01/app/oracle/fast_recovery_area/ORCLCDB/8C28E6F854EB7D
BBE0536210ED0AFDD9/backupset/2019_07_03/o1_mf_nnndf_TAG20190703T
161129_gksnx17n_.bkp tag=TAG20190703T161129 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:07
Finished backup at 03-JUL-19

Starting Control File and SPFILE Autobackup at 03-JUL-19
piece
handle=/u01/app/oracle/fast_recovery_area/ORCLCDB/autobackup/201
9_07_03/o1_mf_s_1012666297_gksnxsjm_.bkp comment=NONE
Finished Control File and SPFILE Autobackup at 03-JUL-19

RMAN>

```

5. Exit RMAN.

```

RMAN> EXIT

Recovery Manager complete.

$

```

6. Set up for this practice by executing the `setup_04_01.sh` script from the `$HOME/labs/DBMod_Recovery` directory. This script creates a new tablespace and a new user. The user 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.

```

$ ./setup_04_01.sh
$

```

7. Cause a failure in the database by executing the `break_04_01.sh` script. Before introducing the failure, the script updates the user table several times. An extended period of time is simulated and several log switches occur. The script saves its output in the `/tmp/break.log` file.

Note: The last digit of the `SALARY` column indicates how many times this table has been updated.

```

$ ./break_04_01.sh

...
SQL>2
      SALARY
-----
      24009

```

```
...  
$
```

8. Log in to SQL*Plus, start the database instance, and attempt to open the ORCLPDB1 pluggable database. Notice the error messages.

```
$ sqlplus / as sysdba  
...  
Connected to an idle instance.  
  
SQL> startup  
...  
Database mounted.  
Database opened.  
  
SQL> ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN;  
ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN  
*  
ERROR at line 1:  
ORA-01157: cannot identify/lock data file 199 - see DBWR trace  
file  
ORA-01110: data file 199:  
'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf'  
SQL>
```

9. Note your data file number and name. In this example, it is data file 199, with a file name of bartbs.dbf. **You must use your own data file number for recovery.** Knowing the data file name is helpful for later correlations.
10. Open a new terminal window. Check the latest DBWR trace file, and then return to the \$HOME/labs/DBMod_Recovery directory.

```
$ cd /u01/app/oracle/diag/rdbms/orclcdb/orclcdb/trace  
$ ls -ltr *dbw*  
...  
-rw-r----- 1 oracle oinstall 11324 Jul  2 20:16  
orclcdb_dbw0_29276.trm  
-rw-r----- 1 oracle oinstall 26047 Jul  2 20:16  
orclcdb_dbw0_29276.trc  
-rw-r----- 1 oracle oinstall   724 Jul  3 15:22  
orclcdb_dbw0_14375.trm  
-rw-r----- 1 oracle oinstall  4152 Jul  3 15:22  
orclcdb_dbw0_14375.trc  
$  
$ tail -20 orclcdb_dbw0_14375.trc  
...  
DDE: Problem Key 'ORA 1110' was flood controlled (0x1) (no  
incident)
```

```

ORA-01110: data file 199:
'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 7
ORA-01157: cannot identify/lock data file 199 - see DBWR trace
file
ORA-01110: data file 199:
'/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 7

$ cd $HOME/labs/DBMod_Recovery
$

```

11. Use the RMAN `LIST FAILURE` command to find more information about the failure. You may see one or more failures listed.

```

$ 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
-----
62          HIGH      OPEN        03-JUL-19    One or more non-
system datafiles are missing

RMAN>

```

12. Use the RMAN `ADVISE FAILURE` command to determine if mandatory manual actions exist, and if automated recovery is available. You may see one or more failures listed.

```

RMAN> ADVISE FAILURE;

Database Role: PRIMARY

List of Database Failures

```

```

=====
Failure ID Priority Status      Time Detected Summary
-----
62          HIGH      OPEN        03-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=43 device type=DISK
analyzing automatic repair options complete

Mandatory Manual Actions
=====
1. If file /u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf
was unintentionally renamed or moved, restore it
2. If you have an export of tablespace BARTBS, then drop and re-
create the tablespace and import the data.
3. Contact Oracle Support Services if the preceding
recommendations cannot be used, or if they do not fix the
failures selected for repair

Optional Manual Actions
=====
no manual actions available

Automated Repair Options
=====
no automatic repair options available
RMAN>

```

The `ADVISE FAILURE` command indicates that mandatory manual actions exist, in addition to optional manual actions and possibly some automated repair options. The mandatory manual actions include manually restoring the `bartbs.dbf` data file. This file name is not identified by file number in this advice, but it does correlate to the file number that you recorded in step 6, and that you found in the trace file content in step 7.

13. Attempt to restore and recover the data file that you identified in the previous steps (in this example the file number is 199, but it might differ in your system). *Be sure to use the correct data file number for your system.*

Note: You should expect this restore and recovery to fail. You should look for errors and information that will help you diagnose the cause of this failure.

```

RMAN> restore datafile 199; /*enter your datafile number*/

Starting restore at 03-JUL-19

```

```

using channel ORA_DISK_1

channel ORA_DISK_1: restoring datafile 00198
input datafile copy RECID=20 STAMP=1012663108 file
name=/u01/app/oracle/backup/ORCLCDB/orclpdb1/bartestdata_D-
ORCLCDB_I-2778750799_TS-BARTBS_FNO-198_5ju5o0q4
destination for restore of datafile 00199:
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf
channel ORA_DISK_1: copied datafile copy of datafile 00198,
elapsed time: 00:00:01
output file
name=/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf RECID=0
STAMP=0
Finished restore at 03-JUL-19

RMAN> recover datafile 199; /*enter your datafile number*/
Starting recover at 03-JUL-19
using channel ORA_DISK_1

starting media recovery

archived log for thread 1 with sequence 1 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_1_gkso7g0z_.arc
archived log for thread 1 with sequence 2 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_2_gkso7hok_.arc
archived log for thread 1 with sequence 3 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_3_gkso7mvp_.arc
archived log for thread 1 with sequence 4 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_4_gkso7pys_.arc
archived log for thread 1 with sequence 5 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_5_gkso7t16_.arc
archived log for thread 1 with sequence 7 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_7_gkso803n_.arc
archived log for thread 1 with sequence 8 is already on disk as
file

```

```

/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_8_gkso832o_.arc
archived log for thread 1 with sequence 9 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_9_gkso9j1w_.arc
RMAN-00571:
=====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS
=====
RMAN-00571:
=====
RMAN-03002: failure of recover command at 07/03/2019 16:23:14
RMAN-06053: unable to perform media recovery because of missing
log
RMAN-06025: no backup of archived log for thread 1 with sequence
6 and starting SCN of 12519818 found to restore
RMAN>

```

The error message indicates that **archive log file 6** is missing.

Note: If you get two RMAN-06025 error messages, focus on the latest one, that is, the one with the highest digits also in the following steps.

In a production system, you would determine if there is another copy of this file, possibly in an OS backup that is unknown to RMAN. If the archive log file can be found and restored, a complete recovery is possible. For this practice assume the archive log file is lost.

Note: The archive log sequence number that you find may be different than the one shown in the example. Make note of your missing archive log sequence number.

14. Return to your first terminal window and use your SQL*Plus session to determine how much data will be lost. Complete recovery is not possible in this situation.

```

SQL> archive log list
Database log mode          Archive Mode
Automatic archival         Enabled
Archive destination        USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 8
Next log sequence to archive 10
Current log sequence        10
SQL>

```

In this example, the current redo log file is sequence number **10**. Log number **6** is missing.

So all the data contained in log files 6 through 10 will be lost.

15. Determine the current SCN by querying V\$DATABASE.

```

SQL> SELECT NAME, DBID, CURRENT_SCN, LOG_MODE, OPEN_MODE FROM
V$DATABASE;

NAME                                DBID CURRENT_SCN LOG_MODE      OPEN_MODE

```



```

-----
ORCLCDB      2778750799      12621213 ARCHIVELOG      READ WRITE

SQL>

```

Note: If you attempt to query the `CURRENT_SCN` column of the `V$DATABASE` view for `orclpdb1`, you will get a value of 0 (zero), and not the last "current" SCN. For example:

```

SQL> ALTER SESSION SET CONTAINER=ORCLPDB1;

Session altered.

SQL> SELECT NAME, DBID, CURRENT_SCN, LOG_MODE, OPEN_MODE FROM
V$DATABASE;

NAME                                DBID CURRENT_SCN LOG_MODE      OPEN_MODE
-----
ORCLCDB      2778750799              0 ARCHIVELOG      READ WRITE

SQL>

```

16. Determine the starting SCN and start time of your missing log (**log 6** in this example). Record the values from the `FIRST_CHANGE#` and `FIRST_TIME` columns.

The value in `FIRST_TIME` can be used to inform users how far back they have to go to recover any transactions that have been lost. Log out of SQL*Plus.

```

SQL> select sequence#, first_change#, first_time, status from
v$archived_log where sequence# = 6 and name is not null; /*enter
your missing log number*/

SEQUENCE# FIRST_CHANGE# FIRST_TIME          S
-----
          6          12519818 03-JUL-19          A

SQL> exit
...
$

```

Note: The SCN was displayed in the RMAN error message, but the first time that this archive log was used, it was not displayed.

Also note that the `V$ARCHIVED_LOG` view contains historic information of prior database incarnations. The `NAME` column of the active database incarnation contains the path and name of the archive log; historic incarnations have a null value. The status `A` is for archived logs, `D` is for deleted ones.

17. Return to your second terminal window. It is recommended to always restore the control file first for incomplete recovery so that potential changes in the data structures are known to RMAN. Perform these steps in your window running RMAN.

- a. Bring the database to `NOMOUNT` state.

```
RMAN> shutdown immediate;

database closed
database dismounted
Oracle instance shut down

RMAN> startup nomount;

connected to target database (not started)
Oracle instance started
...

RMAN>
```

- b. Restore the control file from the autobackup.

```
RMAN> restore controlfile from autobackup;

Starting restore at 03-JUL-19
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=21 device type=DISK

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_03
/o1_mf_s_1012663970_gkslo2lr_.bkp found in the recovery area
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20190703
channel ORA_DISK_1: restoring control file from AUTOBACKUP
/u01/app/oracle/fast_recovery_area/ORCLCDB/autobackup/2019_07_03
/o1_mf_s_1012663970_gkslo2lr_.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 03-JUL-19

RMAN>
```

c. Mount the database.

```
RMAN> alter database mount;

released channel: ORA_DISK_1
Statement processed

RMAN>
```

18. Restore the entire database from a backup that was taken before the missing archive log file with the `RESTORE DATABASE UNTIL SEQUENCE nn` command.

```
RMAN> RESTORE DATABASE UNTIL SEQUENCE 6; /*enter your missing
log number*/

Starting restore at 03-JUL-19
Starting implicit crosscheck backup at 03-JUL-19
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=25 device type=DISK
Crosschecked 36 objects
Finished implicit crosscheck backup at 03-JUL-19

Starting implicit crosscheck copy at 03-JUL-19
using channel ORA_DISK_1
Crosschecked 1 objects
Finished implicit crosscheck copy at 03-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_03
/o1_mf_s_1012663970_gkslo2lr_.bkp

using channel ORA_DISK_1

skipping datafile 5; already restored to file
/u01/app/oracle/oradata/ORCLCDB/pdbseed/system01.dbf
skipping datafile 6; already restored to file
/u01/app/oracle/oradata/ORCLCDB/pdbseed/sysaux01.dbf
skipping datafile 8; already restored to file
/u01/app/oracle/oradata/ORCLCDB/pdbseed/undotbs01.dbf
```

```

skipping datafile 198; already restored to file
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/bartbs.dbf
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: restoring datafile 00003 to
/u01/app/oracle/oradata/ORCLCDB/sysaux01.dbf
channel ORA_DISK_1: restoring datafile 00004 to
/u01/app/oracle/oradata/ORCLCDB/undotbs01.dbf
channel ORA_DISK_1: restoring datafile 00007 to
/u01/app/oracle/oradata/ORCLCDB/users01.dbf
channel ORA_DISK_1: reading from backup piece
/u01/app/oracle/fast_recovery_area/ORCLCDB/backupset/2019_07_02/
o1_mf_nnndf_TAG20190702T163029_gkql0677_.bkp
channel ORA_DISK_1: piece
handle=/u01/app/oracle/fast_recovery_area/ORCLCDB/backupset/2019
_07_02/o1_mf_nnndf_TAG20190702T163029_gkql0677_.bkp
tag=TAG20190702T163029
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:35
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 00013 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/system01.dbf
channel ORA_DISK_1: restoring datafile 00014 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/sysaux01.dbf
channel ORA_DISK_1: restoring datafile 00015 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/undotbs01.dbf
channel ORA_DISK_1: restoring datafile 00016 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb2/users01.dbf
channel ORA_DISK_1: reading from backup piece
/u01/app/oracle/fast_recovery_area/ORCLCDB/8857B419BF707E73E0536
210ED0A54C7/backupset/2019_07_02/o1_mf_nnndf_TAG20190702T163029_
gkqlp6hx_.bkp
channel ORA_DISK_1: piece
handle=/u01/app/oracle/fast_recovery_area/ORCLCDB/8857B419BF707E
73E0536210ED0A54C7/backupset/2019_07_02/o1_mf_nnndf_TAG20190702T
163029_gkqlp6hx_.bkp tag=TAG20190702T163029
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:15
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 00161 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/sysaux01.dbf
channel ORA_DISK_1: restoring datafile 00162 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/undotbs01.dbf
channel ORA_DISK_1: restoring datafile 00163 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/users01.dbf
channel ORA_DISK_1: restoring datafile 00172 to
/u01/app/oracle/oradata/ORCLCDB/orclpdb1/INVENTORY01.DBF
channel ORA_DISK_1: restoring datafile 00197 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_07_02/o1_mf_nnndf_TAG20190702T195850_
gkqfvtyb_.bkp
channel ORA_DISK_1: piece
handle=/u01/app/oracle/fast_recovery_area/ORCLCDB/8C28E6F854EB7D
BBE0536210ED0AFDD9/backupset/2019_07_02/o1_mf_nnndf_TAG20190702T
195850_gkqfvtyb_.bkp tag=TAG20190702T195850
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:15
Finished restore at 03-JUL-19

RMAN>

```

19. Recover the database through your last available log file.

Note: If incremental backups are available they will be applied first, and then the archive logs. The number of log files that need to be applied may vary from the example shown.

```

RMAN> recover database until sequence 6; /*enter your missing
log number*/

Starting recover at 03-JUL-19
using channel ORA_DISK_1

starting media recovery

archived log for thread 1 with sequence 10 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_10_gkqlqfyq_.arc
archived log for thread 1 with sequence 11 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_11_gkqlqg2z_.arc
archived log for thread 1 with sequence 12 is already on disk as
file

```

```
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_12_gkqlwsx3_.arc
archived log for thread 1 with sequence 13 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_13_gkqlyz2w_.arc
archived log for thread 1 with sequence 14 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_14_gkqlyz3b_.arc
archived log for thread 1 with sequence 15 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_15_gkq2bh8q_.arc
archived log for thread 1 with sequence 16 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_16_gkq4sxtg_.arc
archived log for thread 1 with sequence 17 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_17_gkq4sxvx_.arc
archived log for thread 1 with sequence 18 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_18_gkq5g6kc_.arc
archived log for thread 1 with sequence 19 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_19_gkq5mt4k_.arc
archived log for thread 1 with sequence 20 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_20_gkq5mt5n_.arc
archived log for thread 1 with sequence 21 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_21_gkq6ltn6_.arc
archived log for thread 1 with sequence 22 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_22_gkq6ltnw_.arc
archived log for thread 1 with sequence 23 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_23_gkq6pb24_.arc
archived log for thread 1 with sequence 24 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_24_gkq6pt0j_.arc
```

```
archived log for thread 1 with sequence 25 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_25_gkq6zwql_.arc
archived log for thread 1 with sequence 26 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_26_gkq6zyx8_.arc
archived log for thread 1 with sequence 27 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_27_gkq77loz_.arc
archived log for thread 1 with sequence 28 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_28_gkq77oq7_.arc
archived log for thread 1 with sequence 29 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_29_gkq7tc4z_.arc
archived log for thread 1 with sequence 30 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_30_gkq7tc75_.arc
archived log for thread 1 with sequence 31 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_31_gkqbsds5_.arc
archived log for thread 1 with sequence 32 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_32_gkqbt1p7_.arc
archived log for thread 1 with sequence 33 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_33_gkqcscr7k_.arc
archived log for thread 1 with sequence 34 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_34_gkqcscr97_.arc
archived log for thread 1 with sequence 35 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_35_gkqf45nc_.arc
archived log for thread 1 with sequence 36 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/ol_mf_1_36_gkqf45pb_.arc
archived log for thread 1 with sequence 37 is already on disk as
file
```

```

/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_37_gkqfwy6q_.arc
archived log for thread 1 with sequence 38 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_38_gkqfwy8w_.arc
archived log for thread 1 with sequence 39 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_39_gkqgcnnh_.arc
archived log for thread 1 with sequence 40 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_02
/o1_mf_1_40_gkqgcnp1_.arc
archived log for thread 1 with sequence 41 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_41_gkrfmbcx_.arc
archived log for thread 1 with sequence 1 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_1_gkso7g0z_.arc
archived log for thread 1 with sequence 2 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_2_gkso7hok_.arc
archived log for thread 1 with sequence 3 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_3_gkso7mvp_.arc
archived log for thread 1 with sequence 4 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_4_gkso7pys_.arc
archived log for thread 1 with sequence 5 is already on disk as
file
/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_07_03
/o1_mf_1_5_gkso7t16_.arc
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_10_gkqlqfyq_.arc thread=1 sequence=10
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_11_gkqlqg2z_.arc thread=1 sequence=11
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_12_gkqlwsx3_.arc thread=1 sequence=12
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_13_gkqlyz2w_.arc thread=1 sequence=13

```



```
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_14_gkq1yz3b_.arc thread=1 sequence=14
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_15_gkq2bh8q_.arc thread=1 sequence=15
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_16_gkq4sxtg_.arc thread=1 sequence=16
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_17_gkq4sxvx_.arc thread=1 sequence=17
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_18_gkq5g6kc_.arc thread=1 sequence=18
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_19_gkq5mt4k_.arc thread=1 sequence=19
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_20_gkq5mt5n_.arc thread=1 sequence=20
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_21_gkq6ltm6_.arc thread=1 sequence=21
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_22_gkq6ltnw_.arc thread=1 sequence=22
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_23_gkq6pb24_.arc thread=1 sequence=23
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_24_gkq6pt0j_.arc thread=1 sequence=24
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_25_gkq6zwql_.arc thread=1 sequence=25
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_26_gkq6zyx8_.arc thread=1 sequence=26
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_27_gkq77loz_.arc thread=1 sequence=27
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_28_gkq77oq7_.arc thread=1 sequence=28
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_29_gkq7tc4z_.arc thread=1 sequence=29
```

```

archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_30_gkq7tc75_.arc thread=1 sequence=30
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_31_gkqbsds5_.arc thread=1 sequence=31
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_32_gkqbt1p7_.arc thread=1 sequence=32
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_33_gkqcsr7k_.arc thread=1 sequence=33
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_34_gkqcsr97_.arc thread=1 sequence=34
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_35_gkqf45nc_.arc thread=1 sequence=35
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_36_gkqf45pb_.arc thread=1 sequence=36
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_37_gkqfwy6q_.arc thread=1 sequence=37
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_38_gkqfwy8w_.arc thread=1 sequence=38
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_39_gkqgcnnh_.arc thread=1 sequence=39
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_02/o1_mf_1_40_gkqgcnp1_.arc thread=1 sequence=40
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_03/o1_mf_1_41_gkrfmcbx_.arc thread=1 sequence=41
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_03/o1_mf_1_1_gkso7g0z_.arc thread=1 sequence=1
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_03/o1_mf_1_2_gkso7hok_.arc thread=1 sequence=2
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_03/o1_mf_1_3_gkso7mvp_.arc thread=1 sequence=3
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_03/o1_mf_1_4_gkso7pys_.arc thread=1 sequence=4

```

```

archived log file
name=/u01/app/oracle/fast_recovery_area/ORCLCDB/archivelog/2019_
07_03/o1_mf_1_5_gkso7t16_.arc thread=1 sequence=5
media recovery complete, elapsed time: 00:00:24
Finished recover at 03-JUL-19
RMAN>

```

20. Open the database using the RESETLOGS option. Open the ORCLPDB1.

```

RMAN> alter database open resetlogs;

Statement processed

RMAN> ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN;

```

21. In your first terminal window, start a SQL*Plus session as sysdba, and query V\$DATABASE to display the CURRENT_SCN and DBID for both the orclcdb and orclpdb1 containers.

```

$ sqlplus / as sysdba
...
SQL> show pdbs

          CON_ID CON_NAME                                OPEN MODE  RESTRICTED
-----
          2 PDB$SEED                                     READ ONLY  NO
          3 ORCLPDB1                                     READ WRITE NO
          4 ORCLPDB2                                     MOUNTED

SQL> SELECT NAME, DBID, CURRENT_SCN, LOG_MODE, OPEN_MODE FROM
V$DATABASE;

NAME                                DBID  CURRENT_SCN  LOG_MODE      OPEN_MODE
-----
ORCLCDB    2778750799    12521206  ARCHIVELOG    READ WRITE

SQL> ALTER SESSION SET CONTAINER=ORCLPDB1;

Session altered.

SQL> SELECT NAME, DBID, CURRENT_SCN, LOG_MODE, OPEN_MODE FROM
V$DATABASE;

NAME                                DBID  CURRENT_SCN  LOG_MODE      OPEN_MODE
-----
ORCLCDB    2778750799    12521227  ARCHIVELOG    READ WRITE

SQL>

```

22. In your window running RMAN, use the Data Recovery Advisor `LIST FAILURE` command to verify that the failures have been repaired. **Then you must exit so that you can connect to the recovery catalog in the next step.**

```
RMAN> list failure;

Database Role: PRIMARY

no failures found that match specification

RMAN> exit

...
$
```

23. Because the `break_04_01.sh` script removed an archive log to create an issue for your learning purpose, crosscheck all archive logs **connected to the recovery catalog**. Refer to *Course Practice Environment: Security Credentials* for the correct password.

```
$ rman target "/ as sysbackup" catalog rcatowner@rcatpdb
...
connected to target database: ORCLCDB (DBID=2778750799)
recovery catalog database Password: <password>
connected to recovery catalog database

RMAN> CROSSCHECK ARCHIVELOG ALL;

new incarnation of database registered in recovery catalog
starting full resync of recovery catalog
full resync complete
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=4 device type=DISK
validation succeeded for archived log
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCL/archivelog/2018_07_
24/ol_mf_1_25_fogv69kj_.arc RECID=32 STAMP=982347337
validation succeeded for archived log
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCL/archivelog/2018_07_
24/ol_mf_1_26_fogv69m0_.arc RECID=33 STAMP=982347337
validation succeeded for archived log
archived log file
name=/u01/app/oracle/fast_recovery_area/ORCL/archivelog/2018_07_
24/ol_mf_1_27_fogv69n3_.arc RECID=34 STAMP=982347337
Crosschecked 3 objects

RMAN>
```

Note: Your number of objects might be different.

24. Delete obsolete backups, 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:
Type                                Key      Completion Time      Filename/Handle
-----
Datafile Copy                       245      2018-07-18:21:21:01
/u01/backup/orcl/data_D-ORCL_I-1509097982_TS-USERS_FNO-
7_0ot8bbn6
Datafile Copy                       246      2018-07-18:21:21:01
/u01/backup/orcl/data_D-ORCL_I-1509097982_TS-SYSTEM_FNO-
1_0dt8bb10
Datafile Copy                       247      2018-07-18:21:21:01
/u01/backup/orcl/data_D-ORCL_I-1509097982_TS-SYSAUX_FNO-
3_0et8bb1f
...
backup piece
handle=/u01/app/oracle/fast_recovery_area/ORCL/70D3EC602C3B341CE
0532110ED0A1042/backupset/2018_07_24/o1_mf_nnndf_TAG20180724T144
503_foggwsd8_.bkp RECID=43 STAMP=982334745
deleted backup piece
backup piece
handle=/u01/app/oracle/fast_recovery_area/ORCL/autobackup/2018_0
7_24/o1_mf_s_982334764_foggxf0g_.bkp RECID=45 STAMP=982334765
deleted backup piece
backup piece
handle=/u01/app/oracle/fast_recovery_area/ORCL/autobackup/2018_0
7_24/o1_mf_s_982345185_fogs3119_.bkp RECID=52 STAMP=982347185
Deleted 26 objects
RMAN> exit

...
$

```

25. In your SQL*Plus session that is still connected to `orclpdb1`, select the `SALARY` column from one row of the `BAR.BARCOPY` table. The last digit of the salary indicates the number of times the `BARCOPY` table has been updated. The difference between this result and the result in step 2 illustrates that multiple updates could be missing after an incomplete recovery. Exit from SQL*Plus.

```

SQL> select salary from bar.barcopy where rownum < 2;

      SALARY
-----

```

```
24005

SQL> exit

...

$
```

26. Execute the `cleanup_04_01.sh` script from the `$HOME/labs/DBMod_Recovery` directory to remove the new user and tablespace created in this practice. The script saves its output in the `/tmp/cleanup.log` file.

```
$ ./cleanup_04_01.sh

$
```

27. Back up the database. You have a new incarnation of the database and the older backups are obsolete, although there are certain cases where the older backups can be used. A new incarnation of the database was created when the `RESETLOGS` command was executed.

Use the `backup_orclcdb.sh` script to create the backup. The script saves its output in the `/tmp/backup.log` file. This script runs this RMAN command to back up the database:

```
BACKUP DATABASE PLUS ARCHIVELOG DELETE INPUT
```

```
$ ./backup_orclcdb.sh

$
```

28. Keep all terminal windows open for the next practice.

Practice 16-2: Recovering a Table from a Backup

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:

- 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.

```
$ ./setup_04_02.sh
$
```

2. Start a SQL*Plus session and verify your test configuration.

- a. Log in as the `SYS` user.

```
$ sqlplus / as sysdba
...
SQL>
```

- b. Confirm that the database is in `ARCHIVELOG` mode.

```
SQL> SELECT NAME, LOG_MODE, OPEN_MODE FROM V$DATABASE;

NAME          LOG_MODE      OPEN_MODE
-----
ORCLCDB       ARCHIVELOG    READ WRITE
```

```
SQL>
```

- c. Confirm that compatibility is set to 19.0 or higher.

```
SQL> show parameter compatible
```

NAME	TYPE	VALUE
compatible	string	19.0.0
noncdb_compatible	boolean	FALSE

```
SQL>
```

- d. 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>
```

- e. 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
2  FROM DBA_TABLES
3  WHERE OWNER = 'BAR'
4  /
```

TABLE_NAME	TABLESPACE_NAME	STATUS
BARCOPY	BARTBS	VALID

```
SQL>
```


3. 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>
```

4. Confirm or configure autobackup of the control file and perform a level 0 backup.

```
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>

```

5. 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!

6. 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>
```

7. 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;
```

```

Table dropped.

SQL> SELECT NAME, CURRENT_SCN FROM V$DATABASE;

NAME                CURRENT_SCN
-----
ORCLCDB              10631475

SQL>

```

8. 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>

```

9. 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 2239129 <<Your SCN from Step 5
2> 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:

- a. Determines the backup based on the SCN you provide
 - b. Creates an auxiliary instance
 - c. Recovers your tables or table partitions, up to the specified point in time, into this auxiliary instance
 - d. Creates a Data Pump export dump file that contains the recovered objects
 - e. Imports the recovered objects into the target database
 - f. Removes the auxiliary instance
10. 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:
```

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	
			/u01/app/oracle/fast_recovery_area/ORCLCDB/8857B419BF707E73E0536210ED0A54C7/backupset/2019_06_27/o1_mf_nnndf_TAG20190627T160931_gk9tmhpl_.bkp

```
...
```

```
Deleted 9 objects
```

```
RMAN> exit
```

```
...
```

```
$
```

11. 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
...
$
```

12. 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
$
```

13. 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.

```
$ ./setup_05_01.sh
$
```

2. 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'
...
$

```

3. 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>

```

4. 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>

```

5. 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>
```

6. Optionally, confirm that there is no other failure.

```
RMAN> list failure;

Database Role: PRIMARY

no failures found that match specification

RMAN>
```

7. Exit RMAN.

```
RMAN> exit
...
```

Note: This script may end with an error showing that data file `system01.dbf` cannot be opened.

8. 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
```

9. 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
$
```

10. 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.

Practice 18-1: Recovering from the Loss of a Parameter File

Overview

In this practice, you will create an issue by deleting the `initorclpdb.ora` parameter file. After creating the issue, you must restore the parameter file.

Assumptions

A full backup of the database exists. Autobackup of the control file and `SPFILE` is assumed to be configured in the fast recovery area.

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 `orclpdb` database instance.

Tasks

1. Prepare for this practice by executing the `setup_06_01.sh` script from the `$HOME/labs/DBMod_Recovery` directory. This script:
 - Creates a new tablespace and user
 - As the new user creates a table and populates it
 - Performs a backup of the tablespace and then updates the table
 - Saves its output in the `/tmp/setup.log` file

```
$ ./setup_06_01.sh
$
```

2. Cause a failure in the database by executing the `break_06_01.sh` script from the `$HOME/labs/DBMod_Recovery` directory. The script saves its output in the `/tmp/break.log` file.

```
$ ./break_06_01.sh
$
```

3. Attempt to start the database instance. Notice the error messages. Exit from SQL*Plus.

```
$ sqlplus / as sysdba
...
Connected to an idle instance.

SQL> startup
ORA-01078: failure in processing system parameters
LRM-00109: could not open parameter file
'/u01/app/oracle/product/18.3.0/dbhome_1/dbs/initorclpdb.ora'
SQL> exit
...
$
```

4. Start the database by using RMAN.

```
$ rman target "'/ as sysbackup'"
...
connected to target database (not started)

RMAN> startup;
startup failed: ORA-01078: failure in processing system
parameters
LRM-00109: could not open parameter file
'/u01/app/oracle/product/19.3.0/dbhome_1/dbs/initorclcdb.ora'

starting Oracle instance without parameter file for retrieval of
spfile
Oracle instance started
RMAN-00571:
=====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS
=====
RMAN-00571:
=====
RMAN-03002: failure of startup command at 07/01/2019 16:25:01
ORA-00205: error in identifying control file, check alert log
for more info

RMAN>
```

Note: The database has been started with a dummy parameter file to allow the SPFILE to be restored.

5. Restore the SPFILE. Because the database has been started with a dummy parameter file, the location of the autobackup must be specified. In this case, you are using the recovery area and DB_NAME options to specify where the autobackups can be found.

```
RMAN> restore spfile from autobackup recovery area
'/u01/app/oracle/fast_recovery_area' db_name 'orclcdb';

Starting restore at 01-JUL-19
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=180 device type=DISK

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_1012484389_gkn395ot_.bkp found in the recovery area
AUTOBACKUP search with format "%F" not attempted because DBID
was not set
```

```
channel ORA_DISK_1: restoring spfile from AUTOBACKUP
/u01/app/oracle/fast_recovery_area/ORCLCDB/autobackup/2019_07_01
/o1_mf_s_1012484389_gkn395ot_.bkp
channel ORA_DISK_1: SPFILE restore from AUTOBACKUP complete
Finished restore at 01-JUL-19

RMAN>
```

6. Shut down the database instance and restart it with the restored SPFILE.

```
RMAN> shutdown;

Oracle instance shut down

RMAN> startup;

connected to target database (not started)
Oracle instance started
database mounted
database opened
...
RMAN>
```

7. In your second terminal window, execute the `cleanup_06_01.sh` script to clean up from this practice. The script saves its output in the `/tmp/cleanup.log` file.

```
$ ./cleanup_06_01.sh
$
```

8. 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.

Note: Expect substantial output from the backup and delete commands.

```
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 18-2: 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. Autobackup of the control file and `SPFILE` to the fast recovery area 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 `orclpdb` 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
$
```

2. 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
$
```

3. 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
...
$
```

4. 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/orclcdb/orclcdb
3: diag/rdbms/ubff_pitr_orclpdb1_orclcdb/uBFF
4: diag/rdbms/rcatcdb/rcatcdb
5: diag/tnslsnr/edvmlp0/listener
Q: to quit

Please select option: 2

/* 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
```

5. Close the `gedit` window by clicking the close window icon (x) and then exit `adrci`.

```
Please select option: 0
adrci> exit
$
```

6. Check the Data Recovery Advisor for a solution. Preview the suggested solution before executing it.

- a. 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>
```

- b. 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>

```

- c. 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>

```

7. 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>

```

8. 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>

```

9. 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
$

```


10. 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
$
```

2. 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
$
```

3. 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
...
$
```

4. 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
2: diag/rdbms/orclcdb/orclcdb
3: diag/rdbms/ubff_pitr_orclpdb1_orclcdb/uBFF
4: diag/rdbms/rcatcdb/rcatcdb
5: diag/tnslsnr/edvmr1p0/listener
Q: to quit

Please select option: 2

/* 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
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/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.

- 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
$
```

- 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
2. If file
/u01/app/oracle/fast_recovery_area/ORCLCDB/control02.ctl was
unintentionally renamed or moved, restore it
3. If file /u01/app/oracle/oradata/ORCLCDB/control01.ctl was
unintentionally renamed or moved, restore it
4. If this is a standby database, restore the controlfile for a
standby database using RESTORE STANDBY CONTROLFILE FROM
AUTOBACKUP command
5. 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>

```

7. 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>
```

8. 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>

```

9. 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.

10. 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 clause. The database needs to be recovered so that the control file can be synchronized with the data files.

11. 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_gkp2ngh_.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/redola.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/redola
.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/onlineolog/redola
.log thread=1 sequence=7
media recovery complete, elapsed time: 00:00:01
Finished recover at 02-JUL-19

RMAN>

```

12. Open the database with RESETLOGS.

```

RMAN> ALTER DATABASE OPEN RESETLOGS;

Statement processed

RMAN> ALTER PLUGGABLE DATABASE ORCLPDB1 OPEN;

Statement processed

```

13. 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>

```

14. 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;

```

NAME	DBID	CURRENT_SCN	LOG_MODE	OPEN_MODE
ORCLCDB	2778750799	11419154	ARCHIVELOG	READ WRITE

```

SQL> exit
...
$

```

15. Clean up the practice environment by using the `cleanup_06_03.sh` script from the `$HOME/labs/DBMod_Recovery` directory. The script saves its output in the `/tmp/cleanup.log` file.

```

$ ./cleanup_06_03.sh
$

```

16. 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
$
```

2. 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. Use <Ctrl-C> <Return> to exit your login attempt.

```
$ 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.

3. 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*
/u01/app/oracle/product/19.3.0/dbhome_1/dbs/orapwrcatcdb
$
```

The `orapworclcdb.ora` should not be listed because it was deleted by the `break_06_04.sh` script.

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.

4. 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        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.
$

```

5. 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 `<password>` with the correct password.

```

$ orapwd FILE=$ORACLE_HOME/dbs/orapworclcdb ENTRIES=15

Enter password for SYS: <password> <<< Not displayed

```

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.

6. 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 `<password>` with the correct password.

```

$ sqlplus sys@orclcdb as sysdba
...
Enter password: <password> <<< Not displayed
...
SQL>

```

7. Optionally, review the `V$PWFFILE_USERS` view.

```

SQL> desc V$PWFFILE_USERS

```

Name	Null?	Type
USERNAME		VARCHAR2 (128)
SYSDBA		VARCHAR2 (5)
SYSOPER		VARCHAR2 (5)
SYSASM		VARCHAR2 (5)

SYSBACKUP	VARCHAR2 (5)
SYSDG	VARCHAR2 (5)
SYSKM	VARCHAR2 (5)
ACCOUNT_STATUS	VARCHAR2 (30)
PASSWORD_PROFILE	VARCHAR2 (128)
LAST_LOGIN	TIMESTAMP (9) WITH TIME ZONE
LOCK_DATE	DATE
EXPIRY_DATE	DATE
EXTERNAL_NAME	VARCHAR2 (1024)
AUTHENTICATION_TYPE	VARCHAR2 (8)
COMMON	VARCHAR2 (3)
CON_ID	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>

8. Exit SQL*Plus.

```
SQL> exit  
...  
$
```

9. 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/<password>@orclpdb1 as sysdba
...
SQL>
```

2. 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.

3. 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.

- a. 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>
```

- b. Determine the name of the undo tablespace.

```
SQL> SHOW PARAMETER UNDO
```

NAME	TYPE	VALUE
temp_undo_enabled	boolean	FALSE
undo_management	string	AUTO
undo_retention	integer	14400
undo_tablespace	string	UNDOTBS1

```
SQL>
```

- c. Change the `RETENTION GUARANTEE` value of the undo tablespace.

```
SQL> ALTER TABLESPACE UNDOTBS1 RETENTION GUARANTEE;

Tablespace altered.

SQL>
```

- d. 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>
```

- e. 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 instead of `<n>`.

```
SQL> ALTER DATABASE DATAFILE <n> 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.

4. 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
$
```

2. 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
$
```

3. 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`.

- a. Attempt to use the `SHOW RECYCLEBIN` command to view the contents of the recycle bin for `orclpdb1`.

```
$ sqlplus sys/<password>@orclpdb1 as sysdba
...
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.

- b. 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

SQL>
```

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.

- c. 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 characters.

```
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>
```

- d. Restore the object that has the correct columns. Use the correct object name that you found in the previous step.

```
SQL> FLASHBACK TABLE BAR."BIN$jNKlrSHhInLgU6YR7Qqg/Q==$0" TO
BEFORE DROP RENAME TO BAR102A;

Flashback complete.

SQL>
```

4. 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          PHONE_NUMBER          HIRE_DATE JOB_ID
SALARY
-----
COMMISSION_PCT MANAGER_ID DEPARTMENT_ID LOCATION_ID
-----
          100 Steven          King
SKING          515.123.4567          17-JUN-03 AD_PRES
24001
          90

SQL> exit
$
```

5. 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
$
```

2. Determine the current time to the nearest second. Record this as T1.

Note: Your date will be different.

```
$ sqlplus sys/<password>@orclpdb1 as sysdba
...
SQL> select to_char(sysdate, 'yyyy-mm-dd:hh24:mi:ss') from dual;

SYSDATE
-----
2019-07-04:03:00:25

SQL>
```

3. 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
$
```

4. 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>
```

5. 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                24
          110 Accounting             8

11 rows selected.

SQL> exit
...
$
```

6. 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>
```

2. Enable flashback logging and verify.

```
SQL> ALTER DATABASE FLASHBACK ON;

Database altered.

SQL> select FLASHBACK_ON from v$database;
FLASHBACK_ON
-----
YES
```

3. 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

```

4. 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>

```

5. 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>
```

2. View `HR` data. You will use this information for comparison during this practice.
 - a. Determine the sum of the `SALARY` column in the `HR.EMPLOYEES` table.

```
SQL> SELECT sum(salary) FROM hr.employees;

SUM(SALARY)
-----
        691416

SQL>
```

- b. Determine the total number of employees in department 90.

```
SQL> SELECT count(*) FROM hr.employees where department_id=90;

COUNT(*)
-----
          3

SQL>
```

3. 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
  2  set department_id = 90
  3  where job_id = 'IT_PROG'
  4  /

5 rows updated.

SQL> update hr.employees e
  2  set salary = least(e.salary,
  3  (select (min_salary + max_salary)/2 * 1.10
  4  from hr.jobs j
  5  where j.job_id = e.job_id))
  6  where job_id not like 'AD_%'
  7  /

103 rows updated.

SQL>
```

4. Commit your data and determine the current SCN.

```
SQL> COMMIT;

Commit complete.

SQL> SELECT current_scn FROM v$database;

CURRENT_SCN
-----
32886103

SQL>
```

5. Query the data in the `HR` schema again and compare the results to the values you received in the queries in step 2.

- a. Determine the sum of the `SALARY` column in the `HR.EMPLOYEES` table.

```
SQL> SELECT sum(salary) FROM hr.employees;

SUM(SALARY)
-----
      679092.4

SQL>
```

- b. 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
...
$
```

6. 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.

- a. Close the pluggable database `orclpdb1`.

```
$ sqlplus / as sysdba
...
SQL> alter pluggable database orclpdb1 CLOSE;

Pluggable database altered.
```

- b. 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.
```

- c. 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'
```

- d. Open the pluggable database `orclpdb1` using the `resetlogs` option.

```
SQL> alter pluggable database orclpdb1 open resetlogs;  
  
Pluggable database altered.
```

- e. 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>
```

7. Drop the `RP1` guaranteed restore point. Then exit `SQL*Plus`.

```
SQL> DROP RESTORE POINT rp1;  
  
Restore point dropped.  
  
SQL> exit  
...  
$
```

8. Close all terminal windows open for this practice.

Practices for Lesson 22: Using PDB Snapshots

Practices for Lesson 22

There are no practices for Lesson 22.

Practices for Lesson 23: Database Duplication Overview

Practices for Lesson 23

There are no practices for Lesson 23.

**Practices for Lesson 24:
Creating a Backup-Based
Duplicate Database**

Practices for Lesson 24: Overview

Overview

In these practices, you will duplicate an active container database.

Practice 24-1: Duplicating a Database

Overview

In this practice, you learn how to duplicate an active database. `ORCLCDB` is the active database that will be duplicated or cloned as `DBTEST`. The tasks include the following:

- Prepare the destination for the future `DBTEST` database with Oracle Net connectivity, password file, and a minimal initialization parameter file (`initdbtest.ora`).
- Confirm source database configuration settings.
- Duplicate the `ORCLCDB` database by using RMAN.
- Test access to the cloned database.

Tasks

1. Open a terminal window and use `oraenv` to set the environment variables for the `orclcdb` database.

```
[oracle@host ~]$ . oraenv
ORACLE_SID = [orclcdb] ?
[oracle@host ~]$
```

2. Create new directories for the `dbtest` database.

```
$ cd /u01/app/oracle

$ ls
admin backup checkpoints diag
fast_recovery_area product
audit cfgtoollogs controlfiles_dir donttouch oradata

$ mkdir dbtest_fra

$ ls -l
total 48
drwxr-x--- 6 oracle oinstall 4096 Oct 12 11:06 admin
drwxr-x--- 4 oracle oinstall 4096 Jun 27 2019 audit
drwxr-xr-x 4 oracle oinstall 4096 Oct 12 10:15 backup
drwxr-x--- 4 oracle oinstall 4096 May 8 2019 cfgtoollogs
drwxr-xr-x 2 oracle oinstall 4096 May 8 2019 checkpoints
drwxr-xr-x 3 oracle oinstall 4096 Oct 9 12:18 controlfiles_dir
drwxr-xr-x 2 oracle oinstall 4096 Oct 12 14:35 dbtest_fra
drwxrwxr-x 23 oracle oinstall 4096 May 8 2019 diag
drwxr-xr-x 2 oracle oinstall 4096 Oct 12 10:13 donttouch
drwxr-x--- 4 oracle oinstall 4096 Jun 27 2019
fast_recovery_area
drwxr-x--- 4 oracle oinstall 4096 Jun 27 2019 oradata
```

```

drwxr-xr-x  3 oracle oinstall 4096 May  8 2019 product

$ cd dbtest_fra

$ mkdir orclcdb

$ cd ../oradata

$ ls
ORCLCDB  RCATCDB

$ mkdir dbtest

$ ls -l
total 12
drwxr-xr-x 2 oracle oinstall 4096 Oct 12 19:50 dbtest
drwxr-x--- 5 oracle oinstall 4096 May  8 01:43 ORCLCDB
drwxr-x--- 4 oracle oinstall 4096 Jun 27 14:49 RCATCDB

$

```

3. Prepare Oracle Net connectivity by using the Oracle Net Manager utility. Add a service named `dbtest` to the `tnsnames.ora` file.

- a. Launch the Oracle Net Manager utility.

```
$ netmgr
```

- b. Expand Local in the navigation pane, click Service Naming, then click +.
- c. Enter `dbtest` in the Net Service Name field and click Next.
- d. Select TCP for the protocol and click Next.
- e. Enter your host's fully qualified domain name (for example, `edvmr1p0.us.oracle.com`) in the Host name field and click Next.
- f. Enter `dbtest` in the Service Name field, click the Connection Type pull down list and choose Dedicated. Click Next, then Finish.
- g. Click Listeners in the navigation pane, then click +. Select LISTENER from the list. Choose Database Services from the pull down list, then click Add Database and enter `dbtest` in the Global Database Name field and `dbtest` in the SID field. Click File and select Save Network Configuration.
- h. Exit Net Manager utility.

4. View the DBTEST entry in the \$ORACLE_HOME/network/admin/tnsnames.ora file. Ensure the listener recognizes the database service.

```
$ cat $ORACLE_HOME/network/admin/tnsnames.ora
# tnsnames.ora Network Configuration File:
/u01/app/oracle/product/19.3.0/dbhome_1/network/admin/tnsnames.o
ra
# Generated by Oracle configuration tools.

...
DBTEST =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST =
edvmrlp0.us.oracle.com) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = dbtest)
    )
  )

...

$ lsnrctl reload

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 12-OCT-
2019 20:18:53

Copyright (c) 1991, 2019, Oracle.

...

$ lsnrctl status

...
Service "dbtest" has 1 instance(s).
  Instance "dbtest", status READY, has 1 handler(s) for this
service...
...

$
```

5. Create a password file for the future DBTEST database to allow OS authentication.
 - a. As the oracle OS user, set the environment variables to the dbtest instance.

```
$ . oraenv
ORACLE_SID = [orclcdb] ? dbtest
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/19.3.0/dbhome_1
The Oracle base remains unchanged with value /u01/app/oracle
$
```

- b. Create the \$ORACLE_HOME/dbs/orapwdbtest file by using the orapwd utility. Replace <password> with the password listed for this step in the “Course Practice Environment: Security Credentials” document. Please note the SYS password **must** match between the target (orclcdb) and auxiliary (dbtest) databases in order for the duplication operation to complete.

```
$ orapwd file=$ORACLE_HOME/dbs/orapwdbtest entries=15
password=<password>
$
```

- c. Optionally, confirm that the file exists.

```
$ ls $ORACLE_HOME/dbs/*pwwdbtest*
/u01/app/oracle/product/19.3.0/dbhome_1/dbs/orapwdbtest
$
```

6. In the same directory (\$ORACLE_HOME/dbs) use vi or gedit to create a minimal initialization parameter file named initdbtest.ora with the following entries:

```
db_name=dbtest
remote_login_passwordfile=exclusive
```

```
$ gedit $ORACLE_HOME/dbs/initdbtest.ora
```

Then confirm your content with the cat command.

```
$ cat $ORACLE_HOME/dbs/initdbtest.ora
db_name=dbtest
remote_login_passwordfile=exclusive
$
```

7. Log in to SQL*Plus as SYSDBA, start the dbtest instance in NOMOUNT mode with the initdbtest.ora file. Then exit.

```
$ sqlplus / as sysdba
...
Connected to an idle instance.

SQL> startup NOMOUNT
pfile='/u01/app/oracle/product/19.3.0/dbhome_1/dbs/initdbtest.ora'
ORACLE instance started.
```



```
...

SQL> exit

...

$
```

8. Confirm the ORCLCDB source database configuration settings in SQL*Plus.
 - a. Set your environment variables to point to orclcdb, and then log in to SQL*Plus as SYSBDA.

```
$ . oraenv
ORACLE_SID = [dbtest] ? orclcdb
The Oracle base remains unchanged with value /u01/app/oracle

$ sqlplus / as sysdba
...

SQL> show parameter compatible
```

NAME	TYPE	VALUE
compatible	string	19.0.0
noncdb_compatible	boolean	FALSE

```
SQL>
```

- b. Confirm your backup location and size. If the FRA is smaller than 20G, enlarge it by using the following command:

```
ALTER SYSTEM SET db_recovery_file_dest_size = 20G SCOPE=BOTH;
```

```
SQL> show parameter recovery_f
```

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> ALTER SYSTEM SET db_recovery_file_dest_size = 20G
SCOPE=BOTH;

System altered.

SQL> show parameter recovery_f
```

NAME	TYPE	VALUE
db_recovery_file_dest	string	/u01/app/oracle/fast_recovery_area
db_recovery_file_dest_size	big integer	20G
remote_recovery_file_dest	string	
SQL>		

- c. Check and see if the database is in ARCHIVELOG mode. If not, place it 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 63
Next log sequence to archive 65
Current log sequence       65

SQL> exit
...
$
```

9. Use RMAN to duplicate the ORCLCDB database as the DBTEST database.

Note that for the target and catalog connections, the service names are used, but NOT for the auxiliary connection. So you must set the environment variables for the dbtest instance. Refer to the "Course Practice Environment: Security Credentials" document for the passwords for this step.

```
$ . oraenv
ORACLE_SID = [orclcdb] ? dbtest
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/19.3.0/dbhome_1
The Oracle base remains unchanged with value /u01/app/oracle

$ echo $ORACLE_SID
dbtest

$ rman target sys@orclcdb auxiliary sys@dbtest
...
target database Password:
connected to target database: ORCLCDB (DBID=2778750799)
auxiliary database Password:
connected to auxiliary database: DBTEST (DBID=1405367885) (not
mounted)
```

```
RMAN>
```

10. Duplicate the ORCLCDB database as the DBTEST database by using the following command:

```
DUPLICATE TARGET DATABASE TO DBTEST FROM ACTIVE DATABASE
SPFILE PARAMETER_VALUE_CONVERT
'/u01/app/oracle/oradata/ORCLCDB','/u01/app/oracle/oradata/dbtest',
'/u01/app/oracle/fast_recovery_area','/u01/app/oracle/dbtest_fra',
'ORCLCDB','DBTEST'
SET DB_RECOVERY_FILE_DEST_SIZE='20G'
SET DB_FILE_NAME_CONVERT='ORCLCDB','DBTEST';
```

```
RMAN> DUPLICATE TARGET DATABASE TO DBTEST FROM ACTIVE DATABASE
SPFILE PARAMETER_VALUE_CONVERT
'/u01/app/oracle/oradata/ORCLCDB','/u01/app/oracle/oradata/dbtest',
'/u01/app/oracle/fast_recovery_area','/u01/app/oracle/dbtest_fra
', 'ORCLCDB','DBTEST'
SET DB_RECOVERY_FILE_DEST_SIZE='20G'
SET DB_FILE_NAME_CONVERT='ORCLCDB','DBTEST';
...
RMAN> EXIT
```

11. Confirm the availability and accessibility of your newly created dbtest database.

```
$ sqlplus / as sysdba
...
SQL> SELECT NAME, CDB, CON_ID FROM V$DATABASE;
```

NAME	CDB	CON_ID
DBTEST	YES	0

```
SQL> select dbid, name, created, open_mode from v$databases;
```

DBID	NAME	CREATED	OPEN_MODE
1405367885	DBTEST	01-JUL-19	READ WRITE

```
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
SQL>
```

12. Exit SQL*Plus.

```
SQL> exit
...
$
```

Practice 24-2: Duplicating a PDB into an Existing CDB

Overview

In this practice, you will clone `ORCLPDB1` from `ORCLCDB` into `DBTEST` by using the `RMAN DUPLICATE` command without the use of a fresh auxiliary instance. Duplicate `ORCLPDB1` as `DBTESTPDB1` in `DBTEST`.

Tasks

1. Before duplicating `ORCLPDB1`, check that the `HR.REGIONS` table contains rows.

```
$ . oraenv
ORACLE_SID = [dbtest] ? orclcdb
The Oracle base remains unchanged with value /u01/app/oracle

$ sqlplus hr@ORCLPDB1
Enter password: password

SQL> SELECT * FROM regions;

REGION_ID REGION_NAME
-----
         1 Europe
         2 Americas
         3 Asia
         4 Middle East and Africa

SQL> EXIT
$
```

2. In the destination CDB, set the initialization parameter `REMOTE_RECOVERY_FILE_DEST` to restore foreign archive logs. Create the directory for the new PDB data files.

```
$ . oraenv
ORACLE_SID = [orclcdb] ? dbtest
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/19.3.0/dbhome_1
The Oracle base remains unchanged with value /u01/app/oracle
$ sqlplus / AS SYSDBA

SQL> ALTER SYSTEM SET
REMOTE_RECOVERY_FILE_DEST='/home/oracle/labs/DBMod_Duplicate'
SCOPE=BOTH;

System altered.
```

```
SQL> EXIT
```

```
$ mkdir /u01/app/oracle/oradata/DBTEST/dbtestpdb1
```

```
$
```

3. Start RMAN and connect to the source ORCLCDB CDB root.

```
$ rman
```

```
RMAN> CONNECT TARGET sys@ORCLCDB
```

```
target database Password: password
```

```
connected to target database: ORCLCDB (DBID=1504077170)
```

```
RMAN>
```

4. Connect to the auxiliary instance DBTEST into which you will duplicate ORCLPDB1 as DBTESTPDB1.

```
RMAN> CONNECT AUXILIARY sys@DBTEST
```

```
auxiliary database Password: password
```

```
connected to auxiliary database: DBTEST (DBID=1938746784)
```

```
RMAN>
```

5. Duplicate ORCLPDB1 as DBTESTPDB1 in DBTEST.

```
RMAN> DUPLICATE PLUGGABLE DATABASE ORCLPDB1 AS DBTESTPDB1  
FROM ACTIVE DATABASE
```

```
DB_FILE_NAME_CONVERT  
( '/u01/app/oracle/oradata/ORCLCDB/orclpdb1/',  
  '/u01/app/oracle/oradata/DBTEST/dbtestpdb1/' );
```

```
...
```

```
RMAN> EXIT
```

```
$
```

Observe that the `REMOTE_RECOVERY_FILE_DEST` initialization parameter was used to restore foreign archive log.

6. Check that the new PDB contains the HR schema tables.

```
$ sqlplus / AS SYSDBA

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
          5 DBTESTPDB1                                    READ WRITE NO

SQL> ALTER SESSION SET CONTAINER=DBTESTPDB1;

Session altered.

SQL> SELECT * FROM hr.regions;

REGION_ID REGION_NAME
-----
          1 Europe
          2 Americas
          3 Asia
          4 Middle East and Africa

SQL>
```

7. Drop DBTESTPDB1 by using the
@\$/HOME/labs/DBMod_Duplicate/drop_pdb1_in_dbtest.sql script.

```
SQL> CONNECT / AS SYSDBA
Connected.
SQL> @$HOME/labs/DBMod_Duplicate/drop_pdb1_in_dbtest.sql
...
$
```

