Practice 12

Performing Recovery Part II

Practice Target

In this practice you will perform further recovery scenarios in `ORADB` database.

Practice Overview

In this practice, you will perform the full recovery procedure to the following scenarios:

- Recovery of datafiles loss by switching to image copies
- Database point-in-time recovery (DBPITR):
  - without opening the database in read only state
  - with opening the database in read only state

Assumptions

This practice assumes the `srv1` appliance is up and running and its database `ORADB` is running in `OPEN` state.

Pre-requisites

Take a snapshot of `srv1` appliance. Name the snapshot "**Practice 12 Start**".

Caution!

Do not implement the practice without creating the snapshot as instructed by the previous step.

Recovery Scenario 4:

Recovery of datafiles loss by switching to image copies

Scenario assumptions:

- The database is running in `ARCHIVELOG` mode
- One or more datafiles are lost
- Image copies backup of the database are available in the FRA
- Recovery Catalog is not being used

Preparing for the scenario

1. Start Putty and login to `srv1` as `oracle`

2. Invoke RMAN with connecting to `ORADB` as target and take a full backup of the database as image copies.

```
rman target /

BACKUP AS COPY DATABASE TAG 'DB_COPY';
ALTER SYSTEM SWITCH LOGFILE;
```

Simulating the Crash

You will use a manual method to simulate a tablespace loss.

3. Retrieve the datafile full name of the `users` tablespace.

```
SELECT FILE#, NAME FROM V$DATAFILE WHERE TS# = (SELECT TS# FROM V$TABLESPACE WHERE
NAME='USERS');
```

4. Delete the users tablespace datafile.

```
host 'rm <data file full name>';
```

5. Verify that the database instance is still up and running.

```
SELECT * FROM V$VERSION;
```

Recovery Actions

6. Validate the database and obtain the datafile number of the lost datafile.

```
VALIDATE DATABASE;
```

7. Issue the following commands on the tablespace to recover it.

This method is much faster than restoring from backupset.

```
ALTER DATABASE DATAFILE <n> OFFLINE;
SWITCH DATAFILE <n> TO COPY;
RECOVER DATAFILE <n>;
ALTER DATABASE DATAFILE <n> ONLINE;
```

8. Verify the new datafile location.

```
SELECT NAME FROM V$DATAFILE WHERE TS# = (SELECT TS# FROM V$TABLESPACE WHERE
NAME='USERS');
```

Clean up

9. Shutdown `srv1` and restore it to the snapshot "**Practice 12 Start**".

Recovery Scenario 5:

Performing database point-in-time recovery (DBPITR) when the recovery point is known

Scenario Target

In this scenario, some data has been mistakenly deleted from the database and after discussing the incident with the management, you decided to rewind the database to the point before deleting the data.

In this scenario, you know roughly at what time the incident happened and therefore you know that if you rewind the database to a point-in-time before that time, the lost data will definitely be recovered.

Note: To rewind a database to a short-period of time in the past, database flashback is the best technology choice for this purpose.

Scenario assumptions:

- The database is running in `ARCHIVELOG` mode

Preparing for the scenario

10. Invoke RMAN with connecting to `ORADB` as target and take a full backup of the database as backupset.

```
rman target /
BACKUP DATABASE TAG 'DB_FULL';
ALTER SYSTEM SWITCH LOGFILE;
```

Simulating the data loss

11. Simulate data loss by performing the following steps:

a. Start SQL*Plus and connect as sysdba to `ORADB`.

b. Create a testing user.

```
CREATE USER TUSER IDENTIFIED BY oracle
DEFAULT TABLESPACE USERS
QUOTA UNLIMITED ON USERS;
GRANT CONNECT, CREATE TABLE TO TUSER;
```

c. Create the following table.

```
CREATE TABLE TUSER.test TABLESPACE USERS AS SELECT TABLE_NAME FROM DBA_TABLES;
```

d. Take a note of the current time.

```
SELECT TO_CHAR(SYSDATE,'YYYY-MM-DD:HH24:MI:SS') FROM DUAL;
```

e. Issue the following command:

```
ALTER SYSTEM SWITCH LOGFILE;
```

f. Drop the table.

This action is the destructive unintentional action that you want to recover from.

```
DROP TABLE tuser.test;
```

g. Switch the redo logfile

```
ALTER SYSTEM SWITCH LOGFILE;
```

Recovery Actions

The business noticed the data loss and reported the incident to you asking you to recover it. They informed you that the incident happened at some time later than the time that you noted. You, therefore, decided to rewind the database to that point.

12. Invoke RMAN and connect to `ORADB` as target

```
rman target /
```

13. Shutdown the database and start it up in `MOUNT` state.

```
SHUTDOWN IMMEDIATE;
STARTUP MOUNT;
```

14. Perform the DBPITR to rewind the database to the time noted earlier. Replace the time in the following code with noted time.

The time format must match the value of `$NLS_DATE_FORMAT`

```
RUN { SET UNTIL TIME '<YYYY-MM-DD:HH24:MI:SS>';
      RESTORE DATABASE;
      RECOVER DATABASE;}
```

15. Open the database using `RESETLOGS` option.

```
ALTER DATABASE OPEN RESETLOGS;
```

16. Check if the data has been recovered.

```
SELECT COUNT(*) FROM tuser.test;
```

17. Retrieve the incarnation history.

The view `V$DATABASE_INCARNATION` provides the incarnation history as well.

```
LIST INCARNATION OF DATABASE;
```

Note

If you have been provided a wrong recovery point that is later than the incident time, you would not see the lost data. In this case, you need to rewind the database to some point-in-time further backward in the past. But because you opened the database in `RESETLOGS` option, it is now running in an incarnation version that is different from its incarnation version at that point.

In that case, you have to issue the command "`RESET DATABASE TO INCARNATION <DBINC>`", just after mounting the database. DBINC is the previous database incarnation can be obtained from the command "`LIST INCARNATION OF DATABASE`".

Clean up

18. Shutdown `srv1` and restore it to the snapshot "**Practice 12 Start**".

Recovery Scenario 6:

Performing database point-in-time recovery (DBPITR) when the recovery point is not definite

Scenario Target

This scenario is similar to the previous scenario. Some data has been mistakenly deleted from the database and you are asked to rewind the database to the point before deleting the data.

In this scenario, the time at which the incident happened is within a wide range of time window. You need to try recovering the database to several points-in-time till you recover the lost data. You technically need to open the database in read-only mode and check whether the lost data has been recovered before you open the database in `RESETLOGS` mode.

Scenario assumptions:

- The database is running in `ARCHIVELOG` mode

Preparing for the scenario

19. Invoke RMAN with connecting to `ORADB` as target and take a full backup of the database as backupset.

```
rman target /
BACKUP DATABASE TAG 'DB_FULL';
ALTER SYSTEM SWITCH LOGFILE;
```

Simulating the data loss

20. Simulate data loss by performing the following steps:

a. Start SQL*Plus and connect as sysdba to `ORADB`.

```
sqlplus / as sysdba
```

b. Create a testing user.

```
CREATE USER TUSER IDENTIFIED BY oracle DEFAULT TABLESPACE USERS QUOTA UNLIMITED ON
USERS;
GRANT CONNECT, CREATE TABLE TO TUSER;
```

c. Create the following table.

```
CREATE TABLE TUSER.test TABLESPACE USERS AS SELECT TABLE_NAME FROM DBA_TABLES;
```

d. Take a note of the current time. We will refer to this point of time as **time1**.

```
SELECT TO_CHAR(SYSDATE,'YYYY-MM-DD:HH24:MI:SS') FROM DUAL;
```

e. Switch the redo logfile.

```
ALTER SYSTEM SWITCH LOGFILE;
```

f. Drop the table.

This action is the destructive unintentional action that you want to recover from.

```
DROP TABLE tuser.test;
```

g. Wait for a few seconds then switch the logfile.

```
ALTER SYSTEM SWITCH LOGFILE;
```

h. Take a note of the current time. We will refer to this point of time as **time2**.

```
SELECT TO_CHAR(SYSDATE,'YYYY-MM-DD:HH24:MI:SS') FROM DUAL;
```

Recovery Actions

You know that the data loss happened sometime between time1 and time2. So, you decided to first recover the database to time2. If the data is not recovered, you will try recovering the database to time1.

21. Invoke RMAN and connect to ORADB as target and take note of the displayed DBID

```
rman target /
```

22. Perform the DBPITR to rewind the database to **time2** by executing the following commands. Replace the time in the code with the value of time2.

```
SHUTDOWN IMMEDIATE
STARTUP NOMOUNT
SET DBID <obtained DBID>;
RESTORE CONTROLFILE FROM AUTOBACKUP;
ALTER DATABASE MOUNT;
RUN { SET UNTIL TIME '<YYYY-MM-DD:HH24:MI:SS>';
      RESTORE DATABASE;
      RECOVER DATABASE;
    }
```

Note: observe with the steps above that we needed to restore the control file to be able to open the database in read only mode. Without that step, you cannot open the database in read only mode.

23. Open the database in read-only mode.

```
ALTER DATABASE OPEN READ ONLY;
```

24. Check if the data has been recovered.

You will observe that the data is not there. Therefore, you decided to rewind the database to time1.

```
SELECT COUNT(*) FROM tuser.test;
```

25. Perform the DBPITR procedure again for **time1**.

```
SHUTDOWN IMMEDIATE
STARTUP NOMOUNT
SET DBID <obtained DBID>;
RESTORE CONTROLFILE FROM AUTOBACKUP;
ALTER DATABASE MOUNT;

RUN { SET UNTIL TIME '<YYYY-MM-DD:HH24:MI:SS>';
      RESTORE DATABASE;
      RECOVER DATABASE;}

ALTER DATABASE OPEN READ ONLY;
```

26. Check if the data has been recovered.

You will observe that the required data has been recovered.

```
SELECT COUNT(*) FROM tuser.test;
```

27. Shutdown the database then open it using `RESETLOGS` option.

```
SHUTDOWN IMMEDIATE;
STARTUP MOUNT;
ALTER DATABASE OPEN RESETLOGS;
```

Clean up

28. Shutdown `srv1` and restore it to the snapshot "**Practice 12 Start**".

29. Start `srv1`.

30. Delete the snapshot afterwards. It is not needed any more.

Summary

In this practice, you performed the full recovery procedure to the following scenarios:

- Recovery of datafiles loss by switching to image copies
- Database point-in-time recovery (DBPITR):
  - without opening the database in read only state
  - with opening the database in read only state