

RMAN

Recovery Manager

1. Introduction

RMAN(Recovery Manager)is an Oracle tool that allows you to backup, copy, restore, and recover datafiles, control files, and archived redo logs. Oracle Corporation recommends RMAN as its preferred method for backup and recovery.

Backup and Recovery with Recovery Manager

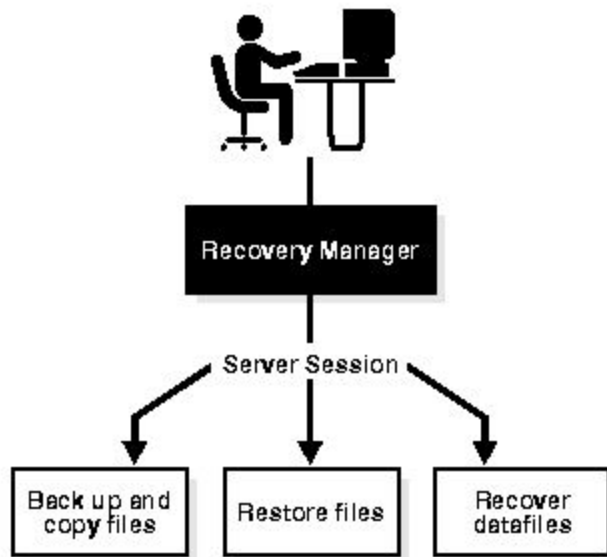


Fig 1.1: RMAN

When we start RMAN, the following operations occur:

- RMAN user session starts on the client.
 - RMAN creates two default server sessions that connects to the target database.
 - If we perform I/O on disk or tape, RMAN requires that we allocate one channel for each disk or device. A channel corresponds to a server session.
-

-
- If we connect to a recovery catalog, RMAN creates a server session on the recovery catalog database.

When we use RMAN to connect to the target database, RMAN uses server sessions to perform the backup and recovery operations through a PL/SQL interface. RMAN physically stores its backups and copies on disk or, if we use media management software, on tape.

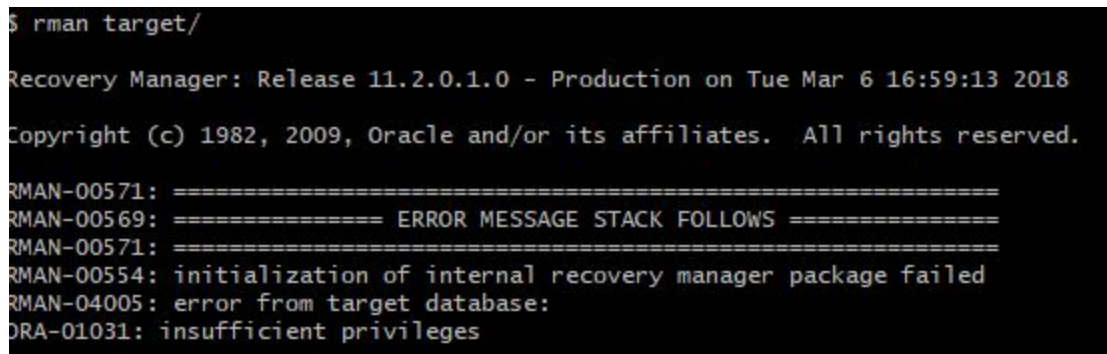
RMAN stores metadata about its backup and recovery operations in the recovery catalog, which is a centralized repository of information, or exclusively in the control file. Typically, the recovery catalog is stored in a separate database. If we do not use a recovery catalog, RMAN uses the control file as its repository of metadata.

2. Connect RMAN

The database that you want to backup or restore is the target database. Target option is used to connect to the target database.

Note: RMAN can only connect to the to user with **sysdba** privilege.

If a user doesn't have sysdba privilege then following error prompts up.



```
$ rman target/  
Recovery Manager: Release 11.2.0.1.0 - Production on Tue Mar 6 16:59:13 2018  
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.  
  
RMAN-00571: =====  
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====  
RMAN-00571: =====  
RMAN-00554: initialization of internal recovery manager package failed  
RMAN-04005: error from target database:  
ORA-01031: insufficient privileges
```

Fig2.1: Error when a user doesn't have sysdba privilege

Solution:

Grant user with sysdba privilege. For example: for user 'saman' grant sysdba privilege by sys.

SQL> Grant sysdba to saman;

Then, Connect to the target by the user

```
(C) 2017 Microsoft Corporation. All rights reserved.  
C:\Users\Saman Munikar>sqlplus sys/saman;  
SQL*Plus: Release 11.2.0.1.0 Production on Tue Mar 6 17:24:47 2018  
Copyright (c) 1982, 2010, Oracle. All rights reserved.  
ERROR:  
ORA-01017: invalid username/password; logon denied  
  
Enter user-name: sys as sysdba  
Enter password:  
  
Connected to:  
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production  
With the Partitioning, OLAP, Data Mining and Real Application Testing options  
  
SQL> show user;  
USER is "SYS"  
SQL> grant sysdba to saman;  
  
Grant succeeded.  
  
SQL> commit;  
  
Commit complete.  
  
SQL>
```

Fig2.2: Grant sysdba to user(saman)

Then, after granting sysdba privilege, connect to the target by that user(saman)

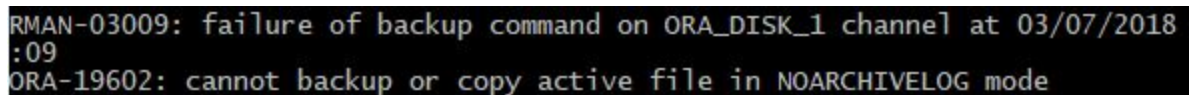
```
C:\Oracle\product\11.2.0\dbhome_1\BIN>rman target saman/saman@saman;  
Recovery Manager: Release 11.2.0.1.0 - Production on Tue Mar 6 17:54:00 2018  
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.  
  
connected to target database: SAMAN (DBID=272096423)  
  
RMAN>
```

Fig2.3: Rman connected saman(user)/saman(password)@saman(service_id)

3. **Backup Database:**

In order to backup the database, we can simply use script as

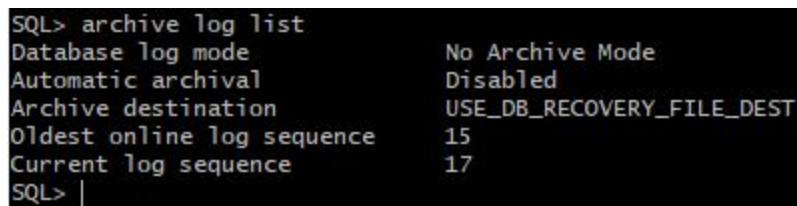
RMAN> backup database;



```
RMAN-03009: failure of backup command on ORA_DISK_1 channel at 03/07/2018
:09
ORA-19602: cannot backup or copy active file in NOARCHIVELOG mode
```

Fig 3.1: Backup Error

To **Check archive log:**



```
SQL> archive log list
Database log mode          No Archive Mode
Automatic archival        Disabled
Archive destination        USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 15
Current log sequence       17
SQL> |
```

We can backup database in both ARCHIVELOG and NOARCHIVELOG mode. However, Oracle has recommended to backup the database in ARCHIVELOG mode. If we backup the database in ARCHIVELOG mode, then we can backup the database while it is open.

Running a database in noarchivelog mode disables the possibility of database recovery as we will not have any archived redo logs to apply to the restored data files and only way of recovery is reverting database to the last full backup and in the same process be ready to lose all changes made after the full backup. This type of backup is called cold backup. This is why oracle does not recommend running database in NOARCHIVELOG mode and does not allow full database backups when database is on noarchivelog mode.

To backup database in NOARCHIVELOG mode:

If a database runs in NOARCHIVELOG mode, then the only valid database backup is a **consistent backup**(A whole database backup that we can open with the RESETLOGS option without performing media recovery).

To backup database in noarchivelog mode, we have to shutdown the database and mount the database but not open. No recovery is required after restoring the database.

```
1      RMAN> run
2      {
3      shutdown immediate;
4      startup mount;
5      shutdown immediate;
6      backup database;
7      ALTER DATABASE OPEN;
8      }
```

i. Start RMAN and connect to a target database.

ii. Shutdown the database consistently and then mount it.

```
RMAN>SHUTDOWN IMMEDIATE;
```

```
RMAN> STARTUP FORCE DBA;
```

```
RMAN>SHUTDOWN IMMEDIATE;
```

```
RMAN> STARTUP MOUNT;
```

iii. Run the backup database command

```
RMAN> BACKUP DATABASE;
```

Create image copy backups of all data files in the database

```
RMAN> BACKUP AS COPY DATABASE;
```

iv. Open the database and resume normal operation.

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

However, as said before, Oracle recommends to backup database in archive log mode.

To alter the archive mode of the database we use ALTER DATABASE statement.

Following steps are carried out to change the database archiving mode from **NOARCHIVELOG to ARCHIVELOG mode**:

a) Shut down the database instance

```
RMAN> SHUTDOWN
```

An open database must first be closed and any associated instances must be shutdown before you can switch the database archiving mode.

b) Backup the database

It's a good idea to create a backup of existing NOARCHIVELOG mode database using exp data pump command.

c) Start a new instance and mount but do not open the database

```
RMAN> STARTUP MOUNT
```

To enable or disable archiving, the database must be mounted but not open.

d) Change the database archiving mode. Then open the database for normal operations.

```
RMAN> ALTER DATABASE ARCHIVELOG;
```

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

e) *Shut down the database:*

RMAN> SHUTDOWN IMMEDIATE

f) *Backup database:*

RMAN> BACKUP DATABASE;

If everything went well, then a control file **O1_MF_NCSNF_TAG20180307T125424_F9Z44JVD_.BKP** (here) with the backup datafiles

O1_MF_NNNDP_TAG20180307T125424_F9Z416LY_.BKP (here) is created.

```
RMAN> backup database;
Starting backup at 07-MAR-18
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=137 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=C:\ORACLE\ORADATA\SAMAN\SYSTEM01.DBF
input datafile file number=00002 name=C:\ORACLE\ORADATA\SAMAN\SYSAUX01.DBF
input datafile file number=00003 name=C:\ORACLE\ORADATA\SAMAN\EXAMPLE01.DBF
input datafile file number=00004 name=C:\ORACLE\ORADATA\SAMAN\UNDOTBS01.DBF
input datafile file number=00005 name=C:\ORACLE\ORADATA\SAMAN\TEST_DATAFILE.DBF
input datafile file number=00006 name=C:\ORACLE\ORADATA\SAMAN\SUMINTTEST_DATAFILE.DBF
input datafile file number=00007 name=C:\ORACLE\ORADATA\SAMAN\USERS01.DBF
channel ORA_DISK_1: starting piece 1 at 07-MAR-18
channel ORA_DISK_1: finished piece 1 at 07-MAR-18
piece handle=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\O1_MF_NNNDP_TAG20180307T125424_F9Z416LY_.BKP tag=TAG20180307T125424 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:01:46
channel ORA_DISK_1: starting full datafile backup set
including current control file in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 07-MAR-18
channel ORA_DISK_1: finished piece 1 at 07-MAR-18
piece handle=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\O1_MF_NCSNF_TAG20180307T125424_F9Z44JVD_.BKP tag=TAG20180307T125424 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03
Finished backup at 07-MAR-18
```

Fig 3.2: Backup database

Backing up database in ARCHIVELOG mode:

If a database runs in ARCHIVELOG mode, then we can back up the database when it is open. This backup is called inconsistent backup because redo is required during recovery to bring the database to consistent state.

Start RMAN and connect to the target database.

Run backup database command.

RMAN> BACKUP DATABASE;

And to backup the database plus the archive redo log files to the default backup device:

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

4. List backup:

We can view the backup created by 'list backup' command .

```
RMAN> list backup;
```

```
RMAN> list backup;

List of Backup Sets
=====

BS Key Type LV Size Device Type Elapsed Time Completion Time
-----
1 Full 9.36M DISK 00:00:05 07-MAR-18
   BP Key: 1 Status: AVAILABLE Compressed: NO Tag: TAG20180307T111656
   Piece Name: C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NCSNF_TAG20180307T111656_F9YYBLSZ_.BKP
   Included: Modification time: 07-MAR-18
   SPFILE db_unique_name: SAMAN
   Control File Included: Ckp SCN: 1334894 Ckp time: 07-MAR-18

BS Key Type LV Size Device Type Elapsed Time Completion Time
-----
2 Full 9.36M DISK 00:00:05 07-MAR-18
   BP Key: 2 Status: AVAILABLE Compressed: NO Tag: TAG20180307T123107
   Piece Name: C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NCSNF_TAG20180307T123107_F9Z200GZ_.BKP
   Included: Modification time: 07-MAR-18
   SPFILE db_unique_name: SAMAN
   Control File Included: Ckp SCN: 1359663 Ckp time: 07-MAR-18

BS Key Type LV Size Device Type Elapsed Time Completion Time
-----
3 Full 1.06G DISK 00:01:36 07-MAR-18
   BP Key: 3 Status: AVAILABLE Compressed: NO Tag: TAG20180307T125424
   Piece Name: C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NNNDP_TAG20180307T125424_F9Z416LY_.BKP
   List of Datafiles in backup set 3
   File LV Type Ckp SCN Ckp Time Name
   -----
   1 Full 1360681 07-MAR-18 C:\ORACLE\ORADATA\SAMAN\SYSTEM01.DBF
   2 Full 1360681 07-MAR-18 C:\ORACLE\ORADATA\SAMAN\SYS_AUX01.DBF
   3 Full 1360681 07-MAR-18 C:\ORACLE\ORADATA\SAMAN\UNDOTBS01.DBF
   4 Full 1360681 07-MAR-18 C:\ORACLE\ORADATA\SAMAN\USERS01.DBF
   5 Full 1360681 07-MAR-18 C:\ORACLE\ORADATA\SAMAN\EXAMPLE01.DBF
   6 Full 1360681 07-MAR-18 C:\ORACLE\ORADATA\SAMAN\TEST_DATAFILE.DBF
   7 Full 1360681 07-MAR-18 C:\ORACLE\ORADATA\SAMAN\SUMINTTEST_DATAFILE.DBF

BS Key Type LV Size Device Type Elapsed Time Completion Time
-----
4 Full 9.36M DISK 00:00:02 07-MAR-18
   BP Key: 4 Status: AVAILABLE Compressed: NO Tag: TAG20180307T125424
   Piece Name: C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NCSNF_TAG20180307T125424_F9Z44JVD_.BKP
   Included: Modification time: 07-MAR-18
   SPFILE db_unique_name: SAMAN
   Control File Included: Ckp SCN: 1360728 Ckp time: 07-MAR-18
```

Fig4.1: List backup

5. Delete backup:

We can delete backup by using the script

RMAN> delete backup;

```
RMAN> delete backup;
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=137 device type=DISK

List of Backup Pieces
BP Key   BS Key   Pc# Cp# Status      Device Type Piece Name
-----
1        1        1   1  AVAILABLE  DISK        C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NCSNF_TAG20180307T111656_F9YYBLSZ_.BKP
2        2        1   1  AVAILABLE  DISK        C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NCSNF_TAG20180307T123107_F9Z200GZ_.BKP
3        3        1   1  AVAILABLE  DISK        C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NNNDP_TAG20180307T125424_F9Z416LY_.BKP
4        4        1   1  AVAILABLE  DISK        C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NCSNF_TAG20180307T125424_F9Z443VD_.BKP

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

Fig 5.1: Delete Backup

6. Recover the deleted database:

To recover the deleted database, we should follow following steps:

- a) *Shutdown immediate;*
- b) *Startup nomount;*
- c) *Restore controlfile from 'path\to\controlfile';*
- d) *Alter database mount;*
- e) *Restore database;*
- f) *Recover database;*
- g) *Alter database open resetlogs;*

First we shutdown all the instances

```
RMAN> shutdown immediate
using target database control file instead of recovery catalog
database closed
database dismounted
Oracle instance shut down
```

Fig 6.1:Shutdown instance

Then, we delete all the datafiles from C:\Oracle\oradata\saman. This directory contains all the datafiles.

```
Recovery Manager complete.  
c:\Oracle\oradata\saman>del *  
c:\Oracle\oradata\saman\*, Are you sure (Y/N)? y
```

Fig 6.2:Deleting datafiles

Now, to recover the datafiles, first we have to startup the database in nomount and then restore the control file we backed up before i.e
**'c:\oracle\flash_recovery_area\saman\
backupset\2018_03_07\O1_MF_NCSNF_TAG20180307T150344_F9ZCPPK0_.BKP'**

```
c:\Oracle\oradata\saman>rman target saman/saman  
Recovery Manager: Release 11.2.0.1.0 - Production on Wed Mar 7 17:33:44 2018  
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.  
connected to target database (not started)  
RMAN>
```

Fig 6.3: Reconnecting RMAN(note: the database is not started here)

A. *Startup at nomount*

```
RMAN> startup nomount;  
Oracle instance started  
Total System Global Area      1536602112 bytes  
Fixed Size                     2176008 bytes  
Variable Size                 1090522104 bytes  
Database Buffers              436207616 bytes  
Redo Buffers                   7696384 bytes
```

Fig 6.4: Startup nomount

B. *Restore controlfile*

```

RMAN> restore controlfile from 'c:\oracle\flash_recovery_area\saman\backupset\2018_03_07\01_MF_NCSNF_TAG20180307T150344_F9ZCPPK0_.BKP'
2> ;
Starting restore at 07-MAR-18
Using target database control file instead of recovery catalog
Allocated channel: ORA_DISK_1
Channel ORA_DISK_1: SID=133 device type=DISK
Channel ORA_DISK_1: restoring control file
Channel ORA_DISK_1: restore complete, elapsed time: 00:00:03
Output file name=C:\ORACLE\ORADATA\SAMAN\CONTROL01.CTL
Output file name=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\CONTROL02.CTL
Finished restore at 07-MAR-18

```

Fig 6.5: Restore control file

After restoring control file, alter database to mount

C. *Alter database mount:*

```

RMAN> alter database mount;
Database mounted
Released channel: ORA_DISK_1

```

Fig 6.6: Alter database mount

D. *restore database*

Then, Restore the datafile

```

RMAN> restore database;
Starting restore at 07-MAR-18
Starting implicit crosscheck backup at 07-MAR-18
Allocated channel: ORA_DISK_1
Channel ORA_DISK_1: SID=133 device type=DISK
Crosschecked 2 objects
Finished implicit crosscheck backup at 07-MAR-18
Starting implicit crosscheck copy at 07-MAR-18
Using channel ORA_DISK_1
Finished implicit crosscheck copy at 07-MAR-18
Searching for all files in the recovery area
Cataloging files...
Cataloging done
List of Cataloged Files
=====
File Name: C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\ARCHIVELOG\2018_03_07\01_MF_1_18_F9ZCPTOW_.ARC
File Name: C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_ANNNN_TAG20180307T150527_F9ZCPW0Q_.BKP
File Name: C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NCSNF_TAG20180307T150344_F9ZCPPK0_.BKP
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 C:\ORACLE\ORADATA\SAMAN\SYSTEM01.DBF
Channel ORA_DISK_1: restoring datafile 00002 to C:\ORACLE\ORADATA\SAMAN\SYSAU01.DBF
Channel ORA_DISK_1: restoring datafile 00003 to C:\ORACLE\ORADATA\SAMAN\UNDOTBS01.DBF
Channel ORA_DISK_1: restoring datafile 00004 to C:\ORACLE\ORADATA\SAMAN\USERS01.DBF
Channel ORA_DISK_1: restoring datafile 00005 to C:\ORACLE\ORADATA\SAMAN\EXAMPLE01.DBF
Channel ORA_DISK_1: restoring datafile 00006 to C:\ORACLE\ORADATA\SAMAN\TEST_DATAFILE.DBF
Channel ORA_DISK_1: restoring datafile 00007 to C:\ORACLE\ORADATA\SAMAN\SUMINTTEST_DATAFILE.DBF
Channel ORA_DISK_1: reading from backup piece C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NNNDP_TAG20180307T150344_F9ZCMPYP_.BKP
Channel ORA_DISK_1: piece handle=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_07\01_MF_NNNDP_TAG20180307T150344_F9ZCMPYP_.BKP tag=TAG20180307T150344
Channel ORA_DISK_1: restored backup piece 1
Channel ORA_DISK_1: restore complete, elapsed time: 00:02:07
Finished restore at 07-MAR-18

```

Fig 6.7: Restore database

E. *recover database:*

Now, we have to recover the database

```

RMAN> recover database;
Starting recover at 07-MAR-18
using channel ORA_DISK_1
starting media recovery
archived log for thread 1 with sequence 18 is already on disk as file C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\ARCHIVELOG\2018_03_07\01_MF_1_18_F9ZCPTOW_.ARC
archived log file name=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\ARCHIVELOG\2018_03_07\01_MF_1_18_F9ZCPTOW_.ARC thread=1 sequence=18
unable to find archived log
archived log thread=1 sequence=19
RMAN-00571: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571:
RMAN-03002: failure of recover command at 03/07/2018 17:48:13
RMAN-06054: media recovery requesting unknown archived log for thread 1 with sequence 19 and starting SCN of 1364405

```

Fig 6.8:Recover Database

F. *alter database open resetlogs:*

Then, we have to alter the database to open resetlogs

```

RMAN> alter database open resetlogs;
database opened

```

Fig 6.9:Alter database open resetlogs

Finally, It's DONE. You have successfully recovered the database.

```

RMAN> quit

Recovery Manager complete.
c:\Oracle\oradata\saman>rman target saman/saman

Recovery Manager: Release 11.2.0.1.0 - Production on Wed Mar 7 17:52:00 2018
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.
connected to target database: SAMAN (DBID=272096423)

```

Fig 6.10: Successfully recover datafiles and datafiles.

7. Backup type:

There are two type of backup namely: full backup and incremental backup.

7.1. Full Backup:

Full backup creates the complete backup of the database objects. Everything that we performed earlier was a full backup.

7.2. Incremental Backup:

Suppose today i created a backup of the database. This backup is called the Level 0 backup in incremental database. And tomorrow I created another incremental backup called Level 1 backup.

In case of incremental backup, only the files changed from level 0 backup to level 1 backup are backed up. This kind of backup is faster and smaller in size.

The two incremental backups are:

a) Differential Incremental Backup:

It is the default incremental backup. Only the files changed from previous last backup to current backup are backed up.

b) Cumulative Incremental Backup:

In this type of incremental backup, all level backup are backed up from level 0 i.e. level 1 backup is backed up from level 0 again level 2 backup is created from level 0 and again level 3 back up is backed up from level 0 and so on.

To create a level 0 incremental backup

```
RMAN> backup incremental level 0 database;

Starting backup at 08-MAR-18
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=17 device type=DISK
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=00001 name=C:\ORACLE\ORADATA\SAMAN\SYSTEM01.DBF
input datafile file number=00002 name=C:\ORACLE\ORADATA\SAMAN\SYSAUX01.DBF
input datafile file number=00005 name=C:\ORACLE\ORADATA\SAMAN\EXAMPLE01.DBF
input datafile file number=00003 name=C:\ORACLE\ORADATA\SAMAN\UNDOTBS01.DBF
input datafile file number=00006 name=C:\ORACLE\ORADATA\SAMAN\TEST_DATAFILE.DBF
input datafile file number=00007 name=C:\ORACLE\ORADATA\SAMAN\SUMINTTEST_DATAFILE.DBF
input datafile file number=00004 name=C:\ORACLE\ORADATA\SAMAN\USERS01.DBF
channel ORA_DISK_1: starting piece 1 at 08-MAR-18
channel ORA_DISK_1: finished piece 1 at 08-MAR-18
piece handle=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_08\01_MF_NNND0_TAG20180308T131423_FB1SLQF9_.BKP tag=TAG20180308T131423 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:01:35
channel ORA_DISK_1: starting incremental level 0 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 08-MAR-18
channel ORA_DISK_1: finished piece 1 at 08-MAR-18
piece handle=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_08\01_MF_NCSN0_TAG20180308T131423_FB1SOSGO_.BKP tag=TAG20180308T131423 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 08-MAR-18
```

Fig 7.1: Incremental level 0 backup

Now, make some changes in the database.

Here, In the name of change, I dropped the tablespace named sumintest.

```
Enter user-name: saman
Enter password:

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> select * from v$tablespace;

      TS# NAME                                INC BIG FLA ENC
-----
      0 SYSTEM                                YES NO  YES
      1 SYSAUX                                YES NO  YES
      2 UNDOTBS1                              YES NO  YES
      4 USERS                                  YES NO  YES
      3 TEMP                                  NO  NO  YES
      6 EXAMPLE                                YES NO  YES
      7 TEST                                  YES NO  YES
      8 SUMINTEST                              YES NO  YES

8 rows selected.

SQL> drop tablespace sumintest;

Tablespace dropped.

SQL> select * from v$tablespace;

      TS# NAME                                INC BIG FLA ENC
-----
      0 SYSTEM                                YES NO  YES
      1 SYSAUX                                YES NO  YES
      2 UNDOTBS1                              YES NO  YES
      4 USERS                                  YES NO  YES
      3 TEMP                                  NO  NO  YES
      6 EXAMPLE                                YES NO  YES
      7 TEST                                  YES NO  YES

7 rows selected.
```

Fig 7.2: Dropped tablespace

Now, I have created a level 1 backup. In this backup, only the changes from level 0 to level 1 is backed up.


```

RMAN> backup incremental level 1 database;

Starting backup at 08-MAR-18
using channel ORA_DISK_1
channel ORA_DISK_1: starting incremental level 1 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001 name=C:\ORACLE\ORADATA\SAMAN\SYSTEM01.DBF
input datafile file number=00002 name=C:\ORACLE\ORADATA\SAMAN\SYSAUX01.DBF
input datafile file number=00005 name=C:\ORACLE\ORADATA\SAMAN\EXAMPLE01.DBF
input datafile file number=00003 name=C:\ORACLE\ORADATA\SAMAN\UNDOTBS01.DBF
input datafile file number=00006 name=C:\ORACLE\ORADATA\SAMAN\TEST_DATAFILE.DBF
input datafile file number=00004 name=C:\ORACLE\ORADATA\SAMAN\USERS01.DBF
channel ORA_DISK_1: starting piece 1 at 08-MAR-18
channel ORA_DISK_1: finished piece 1 at 08-MAR-18
piece handle=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_08\01_MF_NNND1_TAG20180308T132714_FB1TBSKO_.BKP tag=TAG20180308T132714 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:46
channel ORA_DISK_1: starting incremental level 1 datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 08-MAR-18
channel ORA_DISK_1: finished piece 1 at 08-MAR-18
piece handle=C:\ORACLE\FLASH_RECOVERY_AREA\SAMAN\BACKUPSET\2018_03_08\01_MF_NCSN1_TAG20180308T132714_FB1TDB1Y_.BKP tag=TAG20180308T132714 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:02
Finished backup at 08-MAR-18

```

Fig 7.3: Incremental backup level 1

8. Autobackup:

Having a backup of a control file and server parameter file(spfile) is extremely important. RMAN can automatically back up the control file and spfile whenever the RMAN Repository database data has been updated. This is RMAN autobackup.

By default, autobackup is set to off but we could set to on by:

```
RMAN> configure controlfile autobackup on;
```

```

RMAN> configure controlfile autobackup on;

using target database control file instead of recovery catalog
new RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters are successfully stored

```

Fig 8.1: Configure autobackup on

Or we could set to off using

```
RMAN> configure controlfile autobackup off;
```

9. Backup Retention Policies:

We can use CONFIGURE RETENTION POLICY to create a persistent and automatic backup retention policies.

RMAN considers backups of datafiles and control files as **obsolete**, that is, no longer needed for recovery, according to criteria that we specify in the CONFIGURE command.

We can use REPORT OBSOLETE command to view the obsolete files. And, DELETE OBSOLETE command to delete them.

RMAN can identify the obsolete files, but it does not automatically delete them. We have to use DELETE OBSOLETE command to delete files that are no longer needed to satisfy the retention policy.

Warning: **Obsolete** does not mean **expired**. *Obsolete* means “not needed” whereas *Expired* means “not found”. A backup is considered expired only when RMAN performs a crosscheck and cannot find a file.

Also, for datafile backups in backup sets, RMAN cannot delete the backup set until all of the individual datafile backups within the backup set are obsolete.

There are two mutually exclusive options for implementing a retention policy: **redundancy** and **recovery window**. If no recovery policy is configured by the user, then the REPORT OBSOLETE and DELETE OBSOLETE commands use a default retention policy of REDUNDANCY 1.

To configure a retention policy based on a **recovery window**, we use

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW
```

To configure a retention policy based on **redundancy**, we use

```
RMAN> CONFIGURE RETENTION POLICY TO REDUNDANCY
```

However, we can also **disable the retention policy** completely, meaning that RMAN does not consider any backup to be obsolete.

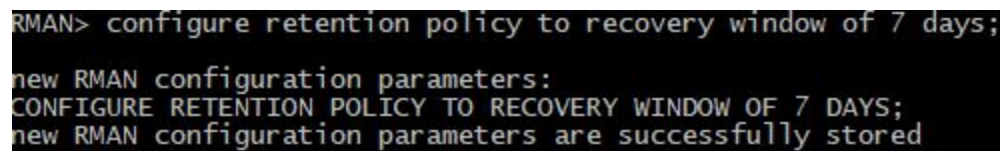
```
RMAN> CONFIGURE RETENTION POLICY TO NONE;
```

If retention policy is configured to none, then REPORT OBSOLETE and DELETE OBSOLETE does not consider backups to be obsolete. As a result, RMAN issues an error when we run REPORT OBSOLETE in retention policy set to none.

8.1. Recovery Window:

A recovery window is a period of time that begins from the present time and extends backward in time to the point of recoverability. For example, If I implement a recovery window of one week, then recovery window will extend back exactly seven days from the current time and we can restore a backup and recover to that point. The script for this example will be:

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 7 DAYS;
```

A screenshot of a terminal window showing the execution of an RMAN command. The text is as follows:

```
RMAN> configure retention policy to recovery window of 7 days;  
new RMAN configuration parameters:  
CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 7 DAYS;  
new RMAN configuration parameters are successfully stored
```

Fig 8.1.1: Retention policy recovery window of 7 days

Not only this, above code will retain one backup that is older than the point of recoverability(i.e. Last backup before the recovery window will also be retained).

To illustrate this assume the retention policy with following aspects:

- Recovery window of 7 days.
- Database backups scheduled every two weeks on
 - January 1
 - January 15
 - January 29
 - February 12
- The database runs in ARCHIVELOG mode, and archived logs are saved on disk only as long as needed for the retention policy.

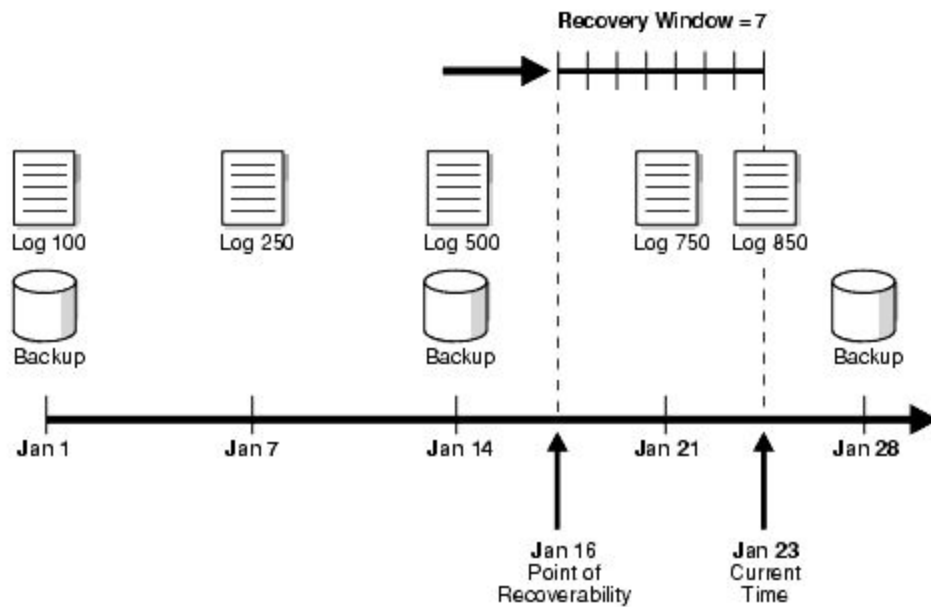


Fig 8.1.2: Recovery window 1st week.

As illustrated in Fig 8.1.1, the recovery window is of 7 days, current day is Jan 23 and the point of recoverability is Jan 16. The January 14 backup and Log 500 to 850 can be used for recovery: however, the logs before 500 and the January 1 backup are obsolete because they are not needed for recovery to a point within the window.

Suppose we take the same scenario a week later,

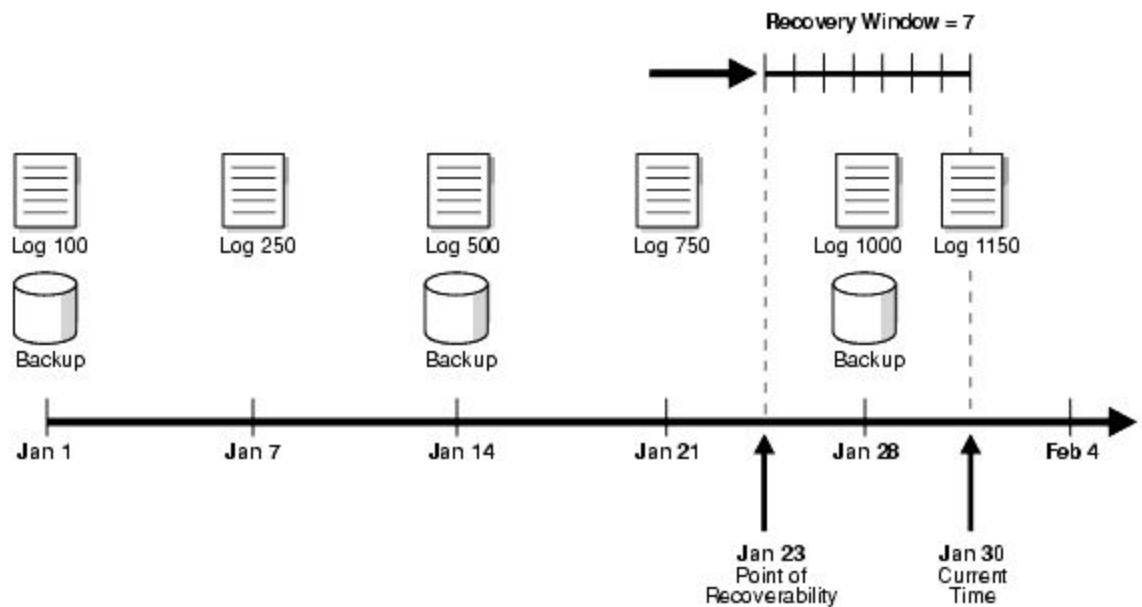


Fig 8.1.3: Recovery Window one week later

In this scenario, the recovery window is same of 7 days, current time is Jan 30 and Point of Recoverability is Jan 23. One thing to notice in this scenario is that, even though backup at Jan 28 exists in the recovery window, Jan 14 backup is still **not obsolete**. Only when the point of recoverability is greater than Jan 23 then Jan 14 backup will be obsolete.

8.2.2. Backup Redundancy:

A redundancy based retention policy specifies how many backups of each datafile must be retained. For example:

```
RMAN> CONFIGURE RETENTION POLICY TO REDUNDANCY 2;
```

```
RMAN> configure retention policy to redundancy 2;

old RMAN configuration parameters:
CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 7 DAYS;
new RMAN configuration parameters:
CONFIGURE RETENTION POLICY TO REDUNDANCY 2;
new RMAN configuration parameters are successfully stored
```

Fig 8.2.1: Retention Policy Redundancy 2

Here, 2 backups of each datafile must be kept.

By default, REDUNDANCY=1

However, Recovery Window is better practice for specifying a retention policy, but it can complicate disk space usage planning because the number of backups that must be kept by the recovery window is not constant and depends on the backup schedule.

Exempting Backups from the Retention Policy:

There may be long term backup case in which we want the backup to persist and not be obsolete for much longer than the time dictated by the retention policy. For example, database backup on the first day of every year: This backup should not be obsolete for 1 year even when number of intermediate backups are conducted within the year. **How can we do this?**

The answer is to use **KEEP** option with the backup command when we create the backup, or the **KEEP** option of the CHANGE command. This KEEP option basically exempts a backup from the retention policy.

KEEP option uses FOREVER value(with LOGS option) or uses data value.

We can specify LOGS option to save the archived logs for a possible incomplete recovery of the long term backup. When logs are specified, all logs more recent than the backup are kept as long as the backup is kept. In other words, KEEP UNTIL TIME... LOGS means that RMAN will keep all logs required to recover the backup as long as the backup is kept. If we specify NOLOGS, then RMAN does not keep the logs required to recover the backup. Note that if we use KEEP UNTIL TIME... with an inconsistent backup, we must use the LOGS option, or that backup will become unusable when the logs required to recover it are deleted as obsolete.

Also, we can use **UNTIL** clause to specify end date, or specify that the backup should be kept **FOREVER**. If we specify UNTIL, then RMAN will not mark the backup as obsolete until after the UNTIL date has passed. Note that it is an error to specify KEEP FOREVER with the LOGS option, as this would require keeping all redo logs forever.

9. Scripting RMAN Operation:

We can use a client side text file containing RMAN commands, exactly as entered in RMAN prompt. We can use any file extension. Use @ command at the RMAN prompt before the path of the file.

```
RMAN> @/path/to/file
```

10. Crosscheck of RMAN Backup:

A user may delete backup pieces from disk using operating system commands, or a tape containing backups may be lost or may become corrupted. To ensure the data about backups in the recovery catalog or control file is synchronized with actual files on disk or in the media management catalog, we perform a **crosscheck**. The CROSSCHECK command operates only on files that are recorded in the RMAN repository.

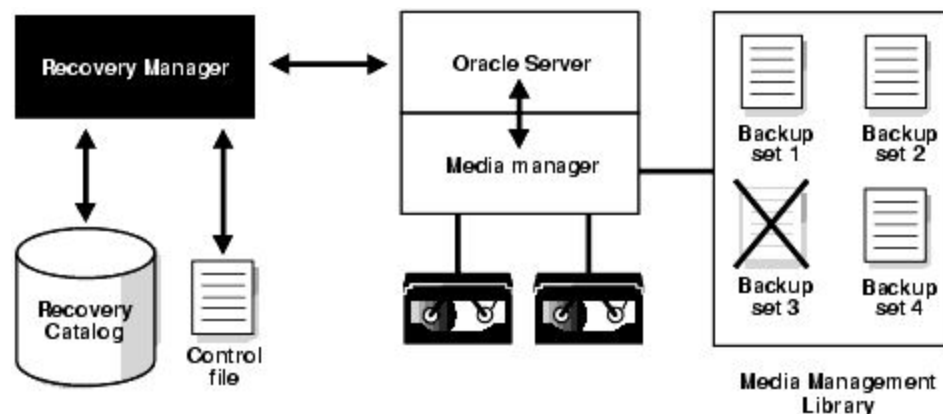


Fig 10.1: Crosschecking the Media Manager

Fig 10.1 illustrates a crosscheck of the media manager. RMAN queries the RMAN repository for the names and locations of the four backup pieces to be checked. RMAN sends this information to the target database server, which queries the media management software about the backups. The media management software then checks its media catalog and reports back to the server that backup set 3 is missing. RMAN updates the status of backup set 3 to EXPIRED in the repository. The record for backup set 3 will now be deleted if you run DELETE EXPIRED.

References:

- [1]. https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRADV89341
- [2]. <https://web.stanford.edu/dept/itss/docs/oracle/10gR2/backup.102/b14191/advmaint002.htm>
- [3]. <https://www.csee.umbc.edu/portal/help/oracle8/server.815/a67773/rmanconc.htm>