

11

Backup and Recovery

There are a thousand excuses for failure but never a good reason. - Mark Twain

In this chapter we will discuss the basics of Oracle Database XE backup and recovery. This chapter introduces Oracle's backup and recovery tool, **Recovery Manager (RMAN)**. We will also discuss a few backup and recovery scenarios. The following are the topics covered in this chapter:

- A brief introduction to backup and recovery
- Recovery Manager
- Connecting to Oracle Database XE using RMAN
- The ARCHIVELOG mode
- Placing a database in the ARCHIVELOG mode
- Backing up a database (the NOARCHIVELOG mode)
- Simulating a database failure (the NOARCHIVELOG mode)
- Restoring the NOARCHIVELOG database
- Configuring the RMAN environment
- Backing up the ARCHIVELOG database
- Simulating a database failure (the ARCHIVELOG mode)
- Restoring ARCHIVELOG database

The full range of backup and recovery techniques is out of the scope of this book. Refer to Oracle documentation for more details.

Introduction to backup and recovery

In information technology, a backup or the process of backing up is making copies of data, which may be used to restore the original after a data loss event (refer to the Backup definition at Wikipedia). Recovering of database is a process a reconstructing the database after data loss.

Recovery Manager

Recovery Manager (RMAN) is Oracle's command-line utility for backing up and recovering an Oracle database. RMAN is fully integrated with Oracle database and is Oracle's recommended tool for backing up and recovering Oracle databases. RMAN is installed automatically with the database.

Connecting to Oracle Database XE using RMAN

Start the RMAN executable (`RMAN.exe`) in the command prompt of your operating system. A RMAN prompt will appear on your screen. Now you can connect to your database using the SYSDBA privilege account as follows:

```
-- connect to the database
RMAN> connect target /

connected to target database: XE (DBID=2655045848)

RMAN> exit
```

Alternatively, you can connect to the database when you start the RMAN client session as follows:

```
-- start RMAN client and connect to the target database

C:\oraclexe\app\oracle\product\11.2.0\server\bin>rman target /

Recovery Manager: Release 11.2.0.2.0 - Production on Wed Apr 4
14:38:53 2012

Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights
reserved.

connected to target database: XE (DBID=2655045848)

RMAN>
```

The ARCHIVELOG mode

Databases can be run in one of two modes – the ARCHIVELOG mode or the NOARCHIVELOG mode. In NOARCHIVELOG mode, the redo log files are reused by the Oracle database engine without being copied to an offline location. In ARCHIVELOG mode, Oracle copies the filled online redo log files to one or more offline locations before they are reused. These redo log files, which are saved offline, are called **archived redo log files**. The ARCH process is responsible for archiving when automatic archiving is enabled. You use the archived redo log files to recover a database and update a standby database.

The ARCHIVELOG mode is very important for mission-critical production databases. Databases can be backed up in the open mode when running in the ARCHIVELOG mode. Also, the ARCHIVELOG mode provides point-in-time recovery. It is generally not necessary for test and development databases.

Placing a database in the ARCHIVELOG mode

~~To place a database in the ARCHIVELOG mode, perform the following steps:~~

1. Create a directory to store archived redo log files.
2. Update the `log_archive_dest_1` initialization parameter to point to the archived redo log file location:

```
SQL> ALTER SYSTEM SET log_archive_dest_1='LOCATION=C:\oraclexe\app\oracle\fast_recovery_area\XE\ARCHIVELOG' SCOPE=both;
```
3. Start the database in MOUNT mode:

```
SQL> STARTUP MOUNT
```
4. Place the database in ARCHIVELOG mode:

```
SQL> ALTER DATABASE ARCHIVELOG;
```
5. Open the database:

```
SQL> ALTER DATABASE OPEN;
```
6. To disable the ARCHIVELOG mode of the database, you would do the following:
 - Start the database in MOUNT mode.
 - Disable the ARCHIVELOG mode:

```
SQL> alter database noarchivelog;
```
 - Start the database in open mode:

```
SQL> alter database open;
```

Backing up a database (the NOARCHIVELOG mode)

You can manually take a backup using the `BACKUP` command in RMAN or use the Oracle Database XE-supplied script. In this section we will use the Oracle Database XE-supplied script to perform a full database backup. Remember, RMAN by default creates backups to disk.

Backing up a database in the `NOARCHIVELOG` mode requires the database to be in a `MOUNT` state. This backup is called a **consistent backup**. When a consistent backup is restored, there is no need to perform any recovery.

Start the backup by running the Oracle Database XE-supplied backup script, located under **Startup | Oracle Database 11g Express Edition | Backup Database** on a Windows machine, and by executing `$ORACLE_HOME/config/scripts/backup.sh` on a Linux environment. The backup script does the following:

- Shuts down the database
- Starts the database in `MOUNT` mode
- Performs backup
- Opens the database for read/write operations

The following screenshot shows RMAN backup in progress on a Windows environment:

```
Warning: Log archiving (ARCHIVELOG mode) is currently disabled. If
you restore the database from this backup, any transactions that take
place between this backup and the next backup will be lost. It is
recommended that you enable ARCHIVELOG mode before proceeding so
that all transactions can be recovered upon restore. See the section
'Enabling ARCHIVELOG Mode...' in the online help for instructions.
Backup with log archiving disabled will shut down and restart the
database. Are you sure [Y/N]? y
Backup in progress...

Recovery Manager: Release 11.2.0.2.0 - Production on Wed Apr 4 15:15:26 2012
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.
connected to target database: XE (DBID=2655045848)

RMAN> set echo on;
2> shutdown immediate;
3> startup mount;
4> configure controlfile autobackup format for device type disk clear;
5> configure retention policy to redundancy 2;
6> configure controlfile autobackup on;
7> sql 'create pfile='C:\ORACLE\XE\APP\ORACLE\PRODUCT\11.2.0\SERVER\DATABASE\SPFILE2\INIT.ORA' from spfile';
8> backup as backupset device type disk database;
9> configure controlfile autobackup off;
10> alter database open;
11> delete noprompt obsolete;
12>
echo set on
using target database control file instead of recovery catalog
```

Now let us check our backup pieces within RMAN, as shown in the following screenshot:

```
RMAN> list backup;
```

List of Backup Sets

```
=====
```

BS Key	Type	LU	Size	Device Type	Elapsed Time	Completion Time
11	Full		824.39M	DISK	00:00:58	04-APR-12
BP Key: 11 Status: AVAILABLE Compressed: NO Tag: TAG20120404T151611						
Piece Name: C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\XE\BACKUPSET\2012_04_04\01_MF_NNNDP_TA						
G20120404T151611_7QRGUDU2_.BKP						
List of Datafiles in backup set 11						
File	LU	Type	Ckp SCN	Ckp Time	Name	
1		Full	573620	04-APR-12	C:\ORACLE\APP\ORACLE\ORADATA\XE\SYSTEM.DBF	
2		Full	573620	04-APR-12	C:\ORACLE\APP\ORACLE\ORADATA\XE\UNDO1BS1.DBF	
3		Full	573620	04-APR-12	C:\ORACLE\APP\ORACLE\ORADATA\XE\SYSAUX.DBF	
4		Full	573620	04-APR-12	C:\ORACLE\APP\ORACLE\ORADATA\XE\USERS.DBF	

```
=====
```

BS Key	Type	LU	Size	Device Type	Elapsed Time	Completion Time
12	Full		9.36M	DISK	00:00:02	04-APR-12
BP Key: 12 Status: AVAILABLE Compressed: NO Tag: TAG20120404T151718						
Piece Name: C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\XE\AUTOBACKUP\2012_04_04\01_MF_S_77972						
8539_7QRGVHQ2_.BKP						
SPFILE Included: Modification time: 04-APR-12						
SPFILE db unique name: XE						
Control File Included: Ckp SCN: 573620 Ckp time: 04-APR-12						

```
RMAN>
```

In the preceding screenshot, you can see that all the database files that belong to our database have been backed up. RMAN groups one or more datafiles into a backup piece, and one or more backup pieces are grouped together and are called a backup set.

Simulating a database failure (the NOARCHIVELOG mode)

Now that we have a valid backup in place, let us simulate a database failure. Let us use the `hr_test` table created earlier in this book for our test. We have five records in the `hr_test` table, as shown in the following screenshot:

```
SQL> select * from hr.hr_test;
```

EMP_NO	EMP_NAME	DATE_OF_B	SALARY
1	Allen Solly	05-JAN-80	500
4	Tom Green	01-FEB-82	80
5	King Lucifer	01-JUN-83	1000
2	Ricky Blake	01-DEC-75	600
3	Karty Sheen	06-MAR-85	700

```
SQL>
SQL>
```

For this test, we will perform the following steps to simulate a database failure:

1. Take a full database backup.
2. Insert a record in the **hr_test** table. (Any changes to the database after the backup will be lost once we restore the database. So, we will lose this record in the recovery process.)
3. Insert the following line of code into **hr_test**:

```
VALUES (6, 'test record', sysdate, 100);
```



```
Shutdown database
```
4. Rename datafiles folder — ORADATA/XE to ORADATA/XE-BACKUP.
5. Create a new empty folder — XE.
6. Start the database now. (Oracle instance starts up in NOMOUNT mode and fails to mount the database, as Oracle is not able to find the control file while mounting the database.)

Restoring the NOARCHIVELOG database

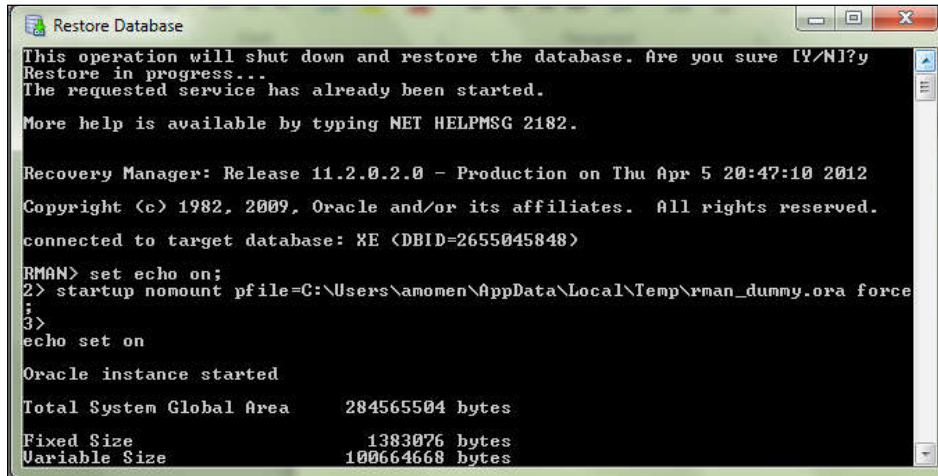
Restoring a backup is a process of bringing the database back to a state before crash. As our database is in the NOARCHIVELOG mode, all changes made after the backup will be lost. We will be able to restore the database to the last backup.

We can restore the database either by using the Oracle Database XE-supplied restore script or by manually entering the restore commands in RMAN command prompt. In this section we will use the Oracle Database XE-supplied script to restore the database.

Start the restore by running the Oracle Database XE-supplied restore script, located under **Startup | Oracle Database 11g Express Edition | Restore Database** on the Window environment, and by executing `$ORACLE_HOME/config/scripts/restore.sh` on Linux. The restore script does the following:

- Restores `spfile` and `controlfile` from `autobackup`
- Restarts the database in `MOUNT` mode
- Restores the database
- Opens the database with the `RESETLOGS` option

The following screenshot shows the RMAN restore process in progress:



```

Restore Database
This operation will shut down and restore the database. Are you sure [Y/N]?y
Restore in progress...
The requested service has already been started.
More help is available by typing NET HELPMSG 2182.

Recovery Manager: Release 11.2.0.2.0 - Production on Thu Apr 5 20:47:10 2012
Copyright (c) 1982, 2009, Oracle and/or its affiliates. All rights reserved.
connected to target database: XE (DBID=2655045848)

RMAN> set echo on;
2> startup nomount pfile=C:\Users\amomen\AppData\Local\Temp\rman_dummy.ora force;
3>
echo set on

Oracle instance started

Total System Global Area      284565504 bytes
Fixed Size                    1383076 bytes
Variable Size                 100664668 bytes

```

Once the RMAN restore procedure completes, our database is ready to use. Log in to the database and query the `hr_test` table for the number of records. You will not find the last inserted test record in the table, because this record did not exist when the backup was taken.

Configuring the RMAN environment

We can use the `SHOW ALL` command to display the current values of RMAN-configured settings for our target database. The following are a few of the values we can configure:

- Database retention policy
- Default device type for backup
- Control file automatic backup
- Enable/disable database encryption

Let us configure automatic backup of controlfile as follows:

```

RMAN> configure controlfile autobackup on;

old RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters are successfully stored

RMAN>

```

This setting will enable the database to take an autobackup of `controlfile` whenever a database backup occurs or the database structure metadata (add/drop of datafiles/tablespaces) in `controlfile` changes.

For more information on other parameters, refer to *Oracle Database Backup and Recovery User's Guide*.

Backing up the ARCHIVELOG database

Let us first place the database in the ARCHIVELOG mode before backing up the database. We have already gone through the procedure of placing the database in the ARCHIVELOG mode earlier in this chapter.

```
SQL> alter system set log_archive_dest_1 = 'LOCATION=C:\oraclexe\app\oracle\fast_recovery_area\XE\ARCHIVELOG' scope=both;
System altered.

SQL>
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
SQL> startup mount
ORACLE instance started.

Total System Global Area  644468736 bytes
Fixed Size                 1385488 bytes
Variable Size             239078384 bytes
Database Buffers          398458880 bytes
Redo Buffers              5545984 bytes
Database mounted.
SQL>
SQL> alter database archivelog;
Database altered.

SQL> alter database open;
Database altered.

SQL> archive log list
Database log mode              Archive Mode
Automatic archival            Enabled
Archive destination            C:\oraclexe\app\oracle\fast_recovery_area\XE\ARCHIVELOG
Oldest online log sequence    1
Next log sequence to archive  1
Current log sequence          1
SQL>
```

Switch a few log files after opening the database using `ALTER SYSTEM SWITCH LOGFILE` and look for new archived redo log files in the `C:\oraclexe\app\oracle\fast_recovery_area\XE\ARCHIVELOG\` folder.

Now our database is ready for the backup. Let us now take a manual backup instead of using the Oracle Database XE-supplied backup script. Log in to the RMAN session, connect to the target database, and run the backup command.

The following is the simple backup command that we will use to back up our database:

```
-- RMAN back command to backup database and archive logs
RMAN> backup database plus archivelog;
```


Alternatively, you can specify the backup location using the `FORMAT` clause. `%U` generates unique filenames for the backup pieces as follows:

```
RMAN> backup database plus archivelog format 'c:\xe\backup_%U';
```

The preceding command does the following:

- Creates a folder under the flash recovery area
- Automatically backs up spfile and controlfile
- Backs up the entire database
- Finally, backs up all the archive log files

The following screenshot shows the backup process:

```

Command Prompt - rman target /

RMAN> backup database plus archivelog;

Starting backup at 07-APR-12
current log archived
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=21 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=1 RECID=1 STAMP=779934596
input archived log thread=1 sequence=2 RECID=2 STAMP=779934598
input archived log thread=1 sequence=3 RECID=3 STAMP=779934598
input archived log thread=1 sequence=4 RECID=4 STAMP=779934599
input archived log thread=1 sequence=5 RECID=5 STAMP=779934599
input archived log thread=1 sequence=6 RECID=6 STAMP=779934620
channel ORA_DISK_1: starting piece 1 at 07-APR-12
channel ORA_DISK_1: finished piece 1 at 07-APR-12
piece handle=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\XE\BACKUPSET\2012_04_07\01_MF_ANNNN_TAG201204
07T003022_7QYR3GKR.BKP tag=TAG20120407T003022 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 07-APR-12

Starting backup at 07-APR-12
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=00002 name=C:\ORACLE\APP\ORACLE\ORADATA\XE\UNDOTBS1.DBF
input datafile file number=00001 name=C:\ORACLE\APP\ORACLE\ORADATA\XE\SYSTEM.DBF
input datafile file number=00004 name=C:\ORACLE\APP\ORACLE\ORADATA\XE\USERS.DBF
input datafile file number=00003 name=C:\ORACLE\APP\ORACLE\ORADATA\XE\SYSAUX.DBF
channel ORA_DISK_1: starting piece 1 at 07-APR-12
channel ORA_DISK_1: finished piece 1 at 07-APR-12
piece handle=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\XE\BACKUPSET\2012_04_07\01_MF_MNNDF_TAG201204
07T003023_7QYR3JQ5.BKP tag=TAG20120407T003023 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:55
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
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 07-APR-12
channel ORA_DISK_1: finished piece 1 at 07-APR-12
piece handle=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\XE\BACKUPSET\2012_04_07\01_MF_MCSNF_TAG201204
07T003023_7QYR5DQ2.BKP tag=TAG20120407T003023 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 07-APR-12

Starting backup at 07-APR-12
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=7 STAMP=779934684
channel ORA_DISK_1: starting piece 1 at 07-APR-12
channel ORA_DISK_1: finished piece 1 at 07-APR-12
piece handle=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\XE\BACKUPSET\2012_04_07\01_MF_ANNNN_TAG201204
07T003124_7QYR5DMZ.BKP tag=TAG20120407T003124 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 07-APR-12

RMAN>

```

Now that we have successfully backed up our database, let us query RMAN to get the backup details. I am listing the backups in summary mode as the verbose mode would generate too much output. However, you may check the verbose on your computers.

```
RMAN> list backup summary;
```

List of Backups										
Key	TY	LU	S	Device	Type	Completion Time	#Pieces	#Copies	Compressed	Tag
9	B	A	A	DISK		07-APR-12	1	1	NO	TAG20120407T003022
10	B	F	A	DISK		07-APR-12	1	1	NO	TAG20120407T003023
11	B	F	A	DISK		07-APR-12	1	1	NO	TAG20120407T003023
12	B	A	A	DISK		07-APR-12	1	1	NO	TAG20120407T003124

```
RMAN>
```

Simulating a database failure (the ARCHIVELOG mode)

Now it's time to start playing with our database. We have recovered our database (NOARCHIVELOG) from a simulated database failure earlier in this chapter. We will repeat a similar approach with our database in the ARCHIVELOG mode.

For this test we will be performing the following steps to simulate a database failure:

1. Take a full database backup.
2. Insert a record in the `hr_test` table:

```
SQL>INSERT into hr_test VALUES (6, 'test record', sysdate, 100);  
SQL> COMMIT;
```
3. Switch the archive logs:

```
SQL> ALTER SYSTEM SWITCH LOGFILE;
```
4. Back up the archive logs:

```
RMAN> backup archivelog all;
```
5. Shut down the database.
6. Rename the datafiles folder — `ORADATA/XE` to `ORADATA/XE-ARCH-BACKUP`.
7. Create a new empty folder — `XE`.
8. Start the database now. (Oracle instance starts up in `NOMOUNT` mode and fails to mount the database, as Oracle is not able to find `controlfile` while mounting the database.)

Restoring the ARCHIVELOG database

The steps involved in recovering an ARCHIVELOG mode database differ from that of the NOARCHIVELOG mode database. The steps involved are as follows:

1. Start the database instance in the NOMOUNT mode.
2. Restore the controlfile.
3. Mount the database.
4. Restore the database.
5. Recover the database.
6. Open the database.
7. Query the hr_test table to confirm the change.

Start the database instance in the NOMOUNT mode using either SQL*Plus or the RMAN client. The next step in the process is to restore controlfile and mount the database. Restore controlfile from the autobackup, as shown in the following screenshot:

```
RMAN> restore controlfile from autobackup;
Starting restore at 08-APR-12
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=10 device type=DISK

recovery area destination: C:\oracle\app\oracle\fast_recovery_area
database name (or database unique name) used for search: XE
channel ORA_DISK_1: AUTOBACKUP C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\XE\AUTOBACKUP\2012_04_08\01_
_MF S.780074967_7R3158LT_.BKP found in the recovery area
channel ORA_DISK_1: looking for AUTOBACKUP on day: 20120408
channel ORA_DISK_1: restoring control file from AUTOBACKUP C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\
XE\AUTOBACKUP\2012_04_08\01_MF S.780074967_7R3158LT_.BKP
channel ORA_DISK_1: control file restore from AUTOBACKUP complete
output file name=C:\ORACLE\APP\ORACLE\ORADATA\XE\CONTROL.DBF
Finished restore at 08-APR-12
RMAN>
```

RMAN searches for a backup of controlfile in the default location, and once located, RMAN restores controlfile to its default location. Now mount the database using the ALTER DATABASE MOUNT command.

The database is now mounted and it is the time to restore the database. The control file has all the information of the backup and when we restore the database, it will read the appropriate backup file and restore the datafiles to their default locations. Restore is a process of copying datafiles from backup location to disk.

The following screenshot shows the **restore database** command:

The next step after restoring the database is to perform database recovery. Recovery is the process of applying redo logs to the database to roll it forward. The following screenshot shows the **recover database** command:

```
RMAN> recover database;
Starting recover at 08-APR-12
using channel ORA_DISK_1
starting media recovery
archived log for thread 1 with sequence 2 is already on disk as file C:\ORACLE\APP\ORACLE\FAST_REC
OVERY_AREA\X\ARCHIVELOGAR0000000002_0780064622.0001
archived log for thread 1 with sequence 3 is already on disk as file C:\ORACLE\APP\ORACLE\FAST_REC
OVERY_AREA\X\ARCHIVELOGAR0000000003_0780064622.0001
archived log for thread 1 with sequence 4 is already on disk as file C:\ORACLE\APP\ORACLE\FAST_REC
OVERY_AREA\X\ARCHIVELOGAR0000000004_0780064622.0001
archived log for thread 1 with sequence 5 is already on disk as file C:\ORACLE\APP\ORACLE\FAST_REC
OVERY_AREA\X\ONLINELOG\01_MF_1_7QYQOGJZ_.LOG
archived log for thread 1 with sequence 6 is already on disk as file C:\ORACLE\APP\ORACLE\FAST_REC
OVERY_AREA\X\ONLINELOG\01_MF_2_7QYQOHKM_.LOG
archived log file name=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\X\ARCHIVELOGAR0000000002_07800646
22.0001 thread=1 sequence=2
archived log file name=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\X\ARCHIVELOGAR0000000003_07800646
22.0001 thread=1 sequence=3
archived log file name=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\X\ARCHIVELOGAR0000000004_07800646
22.0001 thread=1 sequence=4
archived log file name=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\X\ONLINELOG\01_MF_1_7QYQOGJZ_.LOG
thread=1 sequence=5
archived log file name=C:\ORACLE\APP\ORACLE\FAST_RECOVERY_AREA\X\ONLINELOG\01_MF_2_7QYQOHKM_.LOG
thread=1 sequence=6
media recovery complete, elapsed time: 00:00:04
Finished recover at 08-APR-12
RMAN>
```

The **recover database** command applies all archived redo log files and brings the database to the last consistent state. After successfully recovering the database, it's time to open the database. We open the database with the **resetlogs** option, as shown in the following screenshot:

```

RMAN> alter database open resetlogs;
database opened
RMAN>

```

Now that we have successfully restored and recovered the database, it's time to query the **hr_test** table and check for our test record:

```

SQL> select * from hr.hr_test;

```

EMP_NO	EMP_NAME	DATE_OF_B	SALARY
6	test	08-APR-12	100
1	Allen Solly	05-JAN-80	500
4	Tom Green	01-FEB-82	80
5	King Lucifer	01-JUN-83	1000
2	Ricky Blake	01-DEC-75	600
3	Karty Sheen	06-MAR-85	700

```

6 rows selected.
SQL>

```

Our test record with employee number **6** can be seen in the preceding screenshot.

Summary

After completing this chapter, the reader should be able to perform basic Oracle database backups, configure the RMAN environment, change the ARCHIVELOG mode of a database, and recover the database from any media failures.

In this chapter we have used both manual RMAN commands and Oracle-supplied scripts for backup and recovery. At this point, we strongly suggest the reader of this book to refer to Oracle documentation for more details on database backup and recovery.

The next chapter will focus on the performance tuning of the database and database applications. It will cover the best practices in designing efficient database applications and more.

References

- *Oracle Database Backup and Recovery User's Guide*
(http://docs.oracle.com/cd/E11882_01/backup.112/e10642/toc.htm)