EXAM OBJECTIVE: SIMPLIFY FILE MANAGEMENT FOR ALL RECOVERY-RELATED FILES

FLASH RECOVERY AREA OVERVIEW

In Oracle 10g, in order to simplify the management of backup and recovery related files you can configure a special centralized area called a flash recovery area in the database. It is a unified storage location for all recovery related files such as control file backups, archive logs, flashback logs and database backups and activities in an Oracle database.

You can designate a location (typically a directory on disk) and a maximum disk quota for the flash recovery area, and a retention policy to specify your database recoverability goals. You can then direct backup-related activities to store their files in this area.

RECOVERY RELATED FILES

Control File – When the database is created a backup of the control file is created in the flash recovery area.

Archived Log Files – When a flash recovery area is configured, the LOG_ARCHIVE_DEST_10 parameter is automatically set to the flash recovery area. If the LOG_ARCHIVE_DEST_n parameters have not been defined, the default location for archived files is the flash recovery area.

Flashback Logs – Created when flashback has been enabled.

Control File Autobackups – All controlfile backups created by RMAN and autobackups generated by the Oracle database server are by default created in the flash recovery area.

Datafile copies – The BACKUP AS COPY command creates image data file copies in the flash recovery area.

RMAN Files – It is default location that RMAN uses while performing backups, and the location RMAN looks at while performing a restore during a recovery operation.

Parameters that need to be configured:

- **DB_RECOVERY_FILE_DEST_SIZE**
- **DB_RECOVERY_FILE_DEST**

DB_RECOVERY_FILE_DEST_SIZE: This parameter is mandatory to enable the flash recovery area. It determines the maximum amount of space that the flash recovery area is allowed to use. A
DB_RECOVERY_FILE_DEST: This parameter is also mandatory . It identifies the location of the flash recovery area. Leaving this parameter blank disables the flash recovery area.
Examples of setting the parameters dynamically:

```
SQL> ALTER SYSTEM SET
        DB_RECOVERY_FILE_DEST_SIZE=10G
        SCOPE=BOTH;

SQL> ALTER SYSTEM SET
        DB_RECOVERY_FILE_DEST='/home/oracle/oradata/flash'
        SCOPE=BOTH;
```

SPACE MANAGEMENT IN THE FLASH RECOVERY AREA

When RMAN performs a backup and creates a file in the flash recovery area, the list of files that are no longer required on disk is updated. When 85% of the flash recovery area is used a warning is used and a critical warning when the area is 97% full.

When the flash recovery area is full, you can query the **DBA_OUTSTANDING_ALERTS** view to determine what action to take:

```
SQL> SELECT OBJECT_TYPE, MESSAGE_TYPE, MESSAGE_LEVEL,
        REASON, SUGGESTED_ACTION
        FROM DBA_OUTSTANDING_ALERTS;
```

New RMAN Backup Commands

The command to backup all flash recovery files created in the flash recovery area destinations that have not yet been backed to tape is:

```
RMAN> BACKUP RECOVERY AREA;
```

The command that backs up all recovery files on disk that have not been previously backed up to tape is:

```
RMAN> BACKUP RECOVERY FILES;
```

SIMPLIFIED BACKUP

The flash recovery area can be used to store the backups created by RMAN. RMAN will automatically generate the file names in OMF format when the flash

recovery area is used. To backup the entire database to an ASM disk you would issue:

RMAN> BACKUP AS COPY OF DATABASE;

RESTORING THE CONTROL FILE AND SPFILE FROM THE FLASH RECOVERY AREA DURING RECOVERY.

RMAN> RESTORE CONTROLFILE FROM AUTOBACKUP
RECOVERY AREA = '/home/oracle/oradata/flash'

FAST RECOVERY

In Oracle 10g, a fast recovery can be implemented by performing what is known as a "hot restore". In this method you can issue a:

RMAN> SWITCH DATABASE TO COPY;

command to restore the files from the recovery area. No copying of files is done, as in the case of restore in earlier versions. The control file is updated to point to the data files currently located in the flash recovery area.

Note that the use of this command requires the backup files to be image copies and not backupsets.

Information about the flash recovery area can be retrieved by querying the **V$RECOVERY_FILE_DEST** dynamic performance view.

FILE CREATION AND THE FLASH RECOVERY AREA

When you issue the ALTER DATABASE ADD LOGFILE command and do not specify a value for DB_CREATE_ONLINE_LOG_DEST_n then the online redo log is created in the flash recovery area.

When you issue the ALTER DATABASE DROP LOGFILE command, then OMF online redo log files in the flash recovery area are deleted.

The control file can be created in the location specified by CONTROL_FILES or in the location specified by the order of precedence that follows. If DB_CREATE_ONLINE_LOG_DEST_n is specified, then an OMF is created in each directory. If DB_CREATE_ONLINE_LOG_DEST_n is not specified and DB_CREATE_FILE_DEST AND DB_RECOVERY_FILE_DEST are specified they are created in these locations. If the former that is DB_CREATE_FILE_DEST is not specified then it will be created in the DB_RECOVERY_FILE_DEST location.
If none of the above parameters are specified it will be created in an Operating system specific default location.

In the V$CONTROLFILE, V$LOGFILE, V$ARCHIVED_LOG, V$DATAFILE_COPY, V$DATAFILE, V$BACKUP_PIECE and the RMAN tables a new column IS_RECOVERY_DEST_FILE has been introduced. This column takes a value YES if a file of the corresponding kind has been created in the flash recovery area.

EXAM OBJECTIVE: REDUCE RESTORE TIME BY APPLYING INCREMENTAL BACKUPS TO DATA FILE IMAGE COPIES

In Oracle 10g, RMAN can apply incremental backups to data file image copies. With this recovery method, you can use RMAN to recover a copy of a data file, that is roll forward or recover the image copy to the specified point in time by applying the incremental backups.

The benefits of applying incremental backups to data file image copies are:

- Less time for media recovery
- No need to perform a full image copy after incremental backups.

The command you issue to apply the incremental backups to data file image copies is:

```
RMAN> RECOVER COPY OF DATAFILE {n|'file_name'}
```

If there is more than one version of an image copy recorded in the RMAN catalog then RMAN automatically uses the latest version of the image copy.

EXAM OBJECTIVE: SIMPLIFY RECOVERY AFTER OPENING THE DATABASE WITH THE RESETLOGS OPTION

In earlier versions of Oracle, after performing an incomplete recovery the database had to be opened with the RESETLOGS option. This resulted in the creation of a new incarnation of the database. Prior to Oracle 10g, after opening the database with the RESETLOGS option, a complete backup of the database needed to be performed. This was mandatory so that all backups taken henceforth could be used again this backup. Additionally, all previous backups taken before the incomplete recovery are no longer valid against this new incarnation.

Oracle 10g, has introduced a new feature called 'Simplified Recovery through Resetlogs'. This feature allows the use of backups taken prior to the RESETLOGS to be used for recovering the new incarnation. A backup after a RESETLOGS is no longer an important mandatory step. This feature is available when either RMAN or user-managed backup and recovery methods are used.

Also, archived log files created in the earlier incarnation can be used against a previous backup. A new format specification for archived redo log files has been introduced to avoid overwriting archived redo log files with the same sequence

number across all incarnations. It ensures that unique names are constructed for the archived redo log files during RMAN restore and SQL*Plus autorecovery mode. The format specification %r represents the RESETLOGS identifier. The **%r** is included in the default format for the LOG_ARCHIVE_FORMAT initialization parameter.

The COMPATIBILITY parameter has to be set to 10.0.0 for this functionality.

The V$LOG_HISTORY and V$OFFLINE_RANGE records are no longer cleared during the RESETLOGS operation.

EXAM OBJECTIVE: SPEED BACKUP TIMES BY CREATING FASTER INCREMENTAL BACKUPS

In an incremental backup RMAN only backs up those blocks that have changed since a previous backup. Incremental backups of data files, tablespaces or the entire database can be created. When RMAN performs a restore, it checks to see if incremental backups exist and applies them. The entire data file is read during each incremental backup, even if just a very small part of that file has changed since the last incremental backup.

In Oracle 10g, the new **block change tracking** feature, tracks the physical location of all database changes in a new type of file called the change tracking file. The maintenance of this file is fully automatic. A new background process called the **Change Tracking Writer (CTW)** performs the writes to the change tracking writer. The size of the block change tracking file is proportional to:

Fast Incremental Backup can be enabled using Enterprise Manager Console or by issuing commands.

Commands to enable block change tracking are:

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

You can query the V$BLOCK_CHANGE_TRACKING to confirm the status of block change tracking.

```
SQL> SELECT file, status, bytes
       FROM v$block_change_tracking;
```

You can also query the V$BACKUP_DATAFILE view to determine how effective the change tracking data is in minimizing the incremental backup I/O.

EXAM OBJECTIVE: MINIMIZE LOAD REQUIREMENTS BY SPECIFYING LIMITS IN BACKUP TIME WINDOWS

In prior versions of Oracle, RMAN introduced the RATE option on the ALLOCATE or CONFIGURE CHANNEL commands to indicate how many bytes an RMAN channel could read per second per file.

In Oracle 10g, this feature has been extended further to make it even more useful. RMAN can now take an input which is the job time, and computes the throttling speed accordingly. This allows you to control the I/O loads made by backup jobs.

The amount of time to be consumed by a backup job can by specified by using the new DURATION option of the BACKUP command.

Duration <hours> :< minutes> [partial] [minimize <time|load>

where:
duration : specifies the amount of time to run the backup job. It is defined in hours and minutes. The DURATION option now obsoletes the RATE and READRATE options.
partial : If the option is specified, then an error is not reported if the backup is not completed at the end of the specified duration. If it is not specified, and a backup job does not complete in the time duration defined, then by default the backup is cancelled and RMAN terminates with an error. Partial backupsets are aborted and made unusable. If the backup job does not complete on time, whether partial is specified or not, any backup sets that are complete will be cataloged and others must be restarted.
minimize time: Specifies the backup runs at full speed. This may result is the backup completing under the allotted time.
minimize load: With this option RMAN, self monitors its speed and automatically reduces its speed if it detects that it is going to complete in less than the allotted time. The option is not available when writing to tape, as it is desirable to drives the tapes as fast as possible.

- To copy the database within a ten hour time frame. Since you have specified MINIMIZE load, self monitoring of speed is done, and the speed is automatically reduced if it is going to complete in less than ten hours.

  RMAN > BACKUP AS COPY
  DURATION 10:00 MINIMIZE LOAD DATABASE;

- To backup the TOOLS tablespace within four hours or less. In case it is not completed within this stipulated time frame, an error is reported.

```
RMAN> BACKUP DURATION 4:00
FILESPERSET 1 TABLESPACE USERS;
```

- To backup as many files as possible, within a time period of four hours.

```
RMAN> BACKUP PARTIAL DURATION 4:00
FILESPERSET 1 DATABASE;
```

EXAM OBJECTIVE: SAVE STORAGE SPACE THROUGH WRITING COMPRESSED BACKUP SETS

In Oracle 10g you can create compressed backups, to save storage space.
You can create a compressed backup of a database using the command:

```
RMAN> BACKUP AS COMPRESSED BACKUPSET DATABASE;
```

This command requires the COMPATIBILITY parameter to be set to 10.0.0.

A compressed backupset is not usable with pre-Oracle Database 10g databases.
Compression cannot be applied to image copies.

To define a compressed backupset from the Enterprise Manager console, Select **Database Control -> Maintenance tab -> Backup/Recovery region -> Configure Backup Settings link .**