

CLONING ORACLE DATABASE USING BACKUP

PRE-REQUISITES

- A separate Oracle Enterprise Linux (OEL) server with Oracle Database installed.
- Sufficient storage space on the target server.
- Proper network connectivity between source and target servers.
- Ensure the source database is in ARCHIVELOG mode.

ACTIVITY FLOW

- Take source backup
- Move pfile, control file, backup pieces to target server
- Start the instance in mount stage and restore from backup pieces
- Open the DB as source SID
- Rename the DB

TRIGGER BACKUP ON SOURCE

RMAN> backup database plus archivelog delete input;

Restore the Control File

Restoring the control file after taking its backup ensures that the backup metadata is also stored in the control file.

RMAN> restore controlfile to '/tmp/prod_ctrl.ct1';



MOVE FILES TO TARGET SERVER

- The Parameter File (PFILE)
- Edit the PFILE and change the SID except for the DB_NAME parameter (keep it as source).
- Create directories as per the new PFILE.
- `scp /tmp/prod_contrl.ctl target_server:/u01/app/oracle/oradata/testdb/control01.ctl`
- Move the database backup pieces to the target server in the same location as the source.

START CLONING

Export environment variables, connect to RMAN

RMAN> startup mount;

Get the last scn available in the archivelog backup

RMAN> list backup of archivelog all;

Rename the DB redologfiles so they can be created in new location

SQL> alter database rename file /u01/app/oracle/oradata/proddb/redo01.log'
to */u01/app/oracle/oradata/testdb/redo01.log";

RESTORE THE DATAFILES TO NEW LOCATION

run (

set newname for datafile 1 to /u01/app/oracle/oradata/testdb/system01.dbf;

set newname for datafile 2 to '/u01/app/oracle/oradata/testdb/sysaux01.dbf';

set newname for datafile 3 to /u01/app/oracle/oradata/testdb/undotbs01.dbf';

set newname for datafile 4 to /u01/app/oracle/oradata/testdb/users01.dbf';



```
set newname for datafile 5 to /u01/app/oracle/oradata/testdb/example01.dbf";
restore database from tag TAG20250125T090955;
switch datafile all;
recover database until scn 960034;
alter database open resetlogs;
}
```

RENAMING DB AFTER CLONE:

- Connect to the Database
- Take control file backup to trace with resetlogs options.
- Shut down the database.
- Create a PFILE for the new DB ID.
- If locations are the same, only change the DB name.

SQL> shutdown immediate;

SQL> startup nomount;

- Edit the control file in the trace location with the new SID.
- Remove `REUSE` and put `SET`.
- Change `NORESETLOGS` to `RESETLOGS`.
- Remove old control files.
- Create new control files for the instance.
- ALTER DATABASE OPEN RESETLOGS



CREATE CONTROL FILE

**CREATE CONTROLFILE SET DATABASE "PRODDb" RESETLOGS
ARCHIVELOG**

MAXLOGFILES 16

MAXLOGMEMBERS 3

MAXDATAFILES 100

MAXINSTANCES 8

MAXLOGHISTORY 292

LOGFILE

**GROUP 1 '/u01/app/oracle/oradata/testdb/redo01.log' SIZE 50M BLOCKSIZE
512,**

**GROUP 2 '/u01/app/oracle/oradata/testdb/redo02.log' SIZE 50M BLOCKSIZE
512,**

**GROUP 3 '/u01/app/oracle/oradata/testdb/redo03.log' SIZE 50M BLOCKSIZE
512**

DATAFILE

'/u01/app/oracle/oradata/testdb/system01.dbf',

'/u01/app/oracle/oradata/testdb/sysaux01.dbf',

'/u01/app/oracle/oradata/testdb/undotbs01.dbf',

'/u01/app/oracle/oradata/testdb/users01.dbf',

'/u01/app/oracle/oradata/testdb/example01.dbf'

CHARACTER SET WE8MSWIN1252;

=====GOOD LUCK=====

