vahidusefzadeh@gmail.com

useful oracle dba Commands(11g,12C)

Contents

vahidusefzadeh@gmail.com

chapter 1

administration

DATABASE

--CREATE DATABASE

syntax

```
CREATE DATABASE [ database ]  { USER SYS IDENTIFIED BY password  | USER SYSTEM IDENTIFIED BY password
  | CONTROLFILE REUSE  | MAXDATAFILES integer  | MAXINSTANCES integer  | CHARACTER SET charset
  | NATIONAL CHARACTER SET charset  | SET DEFAULT     { BIGFILE | SMALLFILE } TABLESPACE
  | database_logging_clauses  | tablespace_clauses  | set_time_zone_clause
  | [ BIGFILE | SMALLFILE ] USER_DATA TABLESPACE tablespace_name
     DATAFILE datafile_tempfile_spec [, datafile_tempfile_spec ]...  | enable_pluggable_database  }... ;
```

example(Creating a CDB**)**

```
CREATE DATABASE newcdb USER SYS IDENTIFIED BY sys_password
 USER SYSTEM IDENTIFIED BY system_password
 LOGFILE GROUP 1 ('/u01/logs/my/redo01a.log','/u02/logs/my/redo01b.log')
 SIZE 100M BLOCKSIZE 512,
 GROUP 2 ('/u01/logs/my/redo02a.log','/u02/logs/my/redo02b.log')
 SIZE 100M BLOCKSIZE 512,
 GROUP 3 ('/u01/logs/my/redo03a.log','/u02/logs/my/redo03b.log')
 SIZE 100M BLOCKSIZE 512
 MAXLOGHISTORY 1
 MAXLOGFILES 16
 MAXLOGMEMBERS 3
 MAXDATAFILES 1024
 CHARACTER SET AL32UTF8
```

```
NATIONAL CHARACTER SET AL16UTF16

EXTENT MANAGEMENT LOCAL

DATAFILE '/u01/oracle/oradata/newcdb/system01.dbf'

SIZE 700M REUSE AUTOEXTEND ON NEXT 10240K MAXSIZE UNLIMITED

SYSAUX DATAFILE '/u01/oracle/oradata/newcdb/sysaux01.dbf'

SIZE 550M REUSE AUTOEXTEND ON NEXT 10240K MAXSIZE UNLIMITED

 DEFAULT TABLESPACE deftbs

DATAFILE '/u01/oracle/oradata/newcdb/deftbs01.dbf'

SIZE 500M REUSE AUTOEXTEND ON MAXSIZE UNLIMITED

DEFAULT TEMPORARY TABLESPACE tempts1

TEMPFILE '/u01/oracle/oradata/newcdb/temp01.dbf'

SIZE 20M REUSE AUTOEXTEND ON NEXT 640K MAXSIZE UNLIMITED

UNDO TABLESPACE undotbs1

DATAFILE '/u01/oracle/oradata/newcdb/undotbs01.dbf'

SIZE 200M REUSE AUTOEXTEND ON NEXT 5120K MAXSIZE UNLIMITED

ENABLE PLUGGABLE DATABASE

SEED

FILE_NAME_CONVERT = ('/u01/oracle/oradata/newcdb/', '/u01/oracle/oradata/usefdb/')

SYSTEM DATAFILES SIZE 125M AUTOEXTEND ON NEXT 10M MAXSIZE UNLIMITED

SYSAUX DATAFILES SIZE 100M

USER_DATA TABLESPACE usertbs

DATAFILE '/u01/oracle/oradata/usefdb/usertbs01.dbf'

SIZE 200M REUSE AUTOEXTEND ON MAXSIZE UNLIMITED;
```

--ALTER DATABASE

syntax

ALTER DATABASE [database] { startup_clauses | recovery_clauses | database_file_clauses

| logfile_clauses | controlfile_clauses | standby_database_clauses | default_settings_clauses

| instance_clauses | security_clause } ;

examples

ALTER DATABASE ARCHIVELOG;

ALTER DATABASE RECOVER;

ALTER DATABASE SET DEFAULT BIGFILE TABLESPACE;

ALTER DATABASE NO FORCE LOGGING;

ALTER DATABASE OPEN RESETLOGS;

--ALTER SESSION

syntax

ALTER SESSION { ADVISE { COMMIT | ROLLBACK | NOTHING } | CLOSE DATABASE LINK dblink

| { ENABLE | DISABLE } COMMIT IN PROCEDURE | { ENABLE | DISABLE } GUARD

| { ENABLE | DISABLE | FORCE } PARALLEL { DML | DDL | QUERY } [PARALLEL integer]

| { ENABLE RESUMABLE [TIMEOUT integer] [NAME string] | DISABLE RESUMABLE

} | SYNC WITH PRIMARY | alter_session_set_clause } ;

examples

ALTER SESSION DISABLE PARALLEL DDL;

ALTER SESSION ENABLE PARALLEL DML;

ALTER SESSION FORCE PARALLEL DDL PARALLEL 5;

ALTER SESSION FORCE PARALLEL QUERY PARALLEL 4;

ALTER SESSION SET SQL_TRACE TRUE;

ALTER SESSION CLOSE DATABASE LINK linkname;

--ALTER SYSTEM

syntax

ALTER SYSTEM { archive_log_clause | checkpoint_clause | check_datafiles_clause | distributed_recov_clauses

| FLUSH { SHARED_POOL | BUFFER_CACHE | REDO TO target_db_name [[NO] CONFIRM APPLY] }

| end_session_clauses | SWITCH LOGFILE | { SUSPEND | RESUME } | quiesce_clauses

| rolling_migration_clauses | rolling_patch_clauses | security_clauses | shutdown_dispatcher_clause

| REGISTER | SET alter_system_set_clause [alter_system_set_clause]... | RESET alter_system_reset_clause

[alter_system_reset_clause]... | RELOCATE CLIENT client_id } ;

examples

alter system flush shared_pool;

alter system flush buffer_cache;

alter system enable restricted session;

alter system disable restricted session;

alter system disconnect session;

alter system kill session ' sid, serial#' IMMEDIATE;

alter system switch logfile;

ALTER SYSTEM QUIESCE RESTRICTED;

ALTER SYSTEM UNQUIESCE;

ALTER SYSTEM SUSPEND;

ALTER SYSTEM RESUME;

ALTER SYSTEM SHUTDOWN IMMEDIATE 'D002';

ALTER SYSTEM DISABLE DISTRIBUTED RECOVERY;

ALTER SYSTEM ENABLE DISTRIBUTED RECOVERY;

ALTER SYSTEM CHECK DATAFILES;

ALTER SYSTEM REGISTER;

--DROP DATABASE

DROP DATABASE ;

--Startup/Shutdown

STARTUP [FORCE] [RESTRICT] [PFILE=filename] [OPEN [RECOVER] [database] |MOUNT |NOMOUNT]

ALTER DATABASE { MOUNT | OPEN }

ALTER DATABASE OPEN [READ WRITE| READ ONLY]

ALTER SYSTEM ENABLE | DISABLE RESTRICTED SESSION;

vahidusefzadeh@gmail.com

SHUTDOWN [NORMAL | TRANSACTIONAL | IMMEDIATE | ABORT]

--useful query:

```
select a.name "DB Name", e.global_name "Global Name", b.banner "DB Version", c.host_name "Host Name",
c.instance_name "Instance Name" ,c.startup_time "Instance Start Time", decode(c.logins,'RESTRICTED','YES','NO')
"Restricted Mode",a.log_mode  "Archive Log Mode" , decode(a.open_mode,'READ ONLY','YES','NO') "Read Only
Mode" from v$database a, v$version b, v$instance c,global_name e  WHERE b.banner LIKE '%Oracle%'
```

---STARTUP TIME:

```
select instance_name||', up since '|| to_char(startup_time,'DD-MON-YYYY HH24:MI:SS')
start_time,round(sysdate-STARTUP_TIME) "day" from gv$instance;
```

OR

```
select round(sysdate-STARTUP_TIME) "time-start" from v$instance c ;
```

---

parameter file

-- CREATE PFILE

syntax

CREATE PFILE [= 'pfile_name'] FROM { SPFILE [= 'spfile_name'] | MEMORY } ;

example

create pfile='/u02/usef.ora' from spfile;

--CREATE SPFILE

syntax

CREATE SPFILE [= 'spfile_name'] FROM { PFILE [= 'pfile_name'] | MEMORY } ;

example

create spfile from pfile;

CREATE SPFILE FROM MEMORY;

--Recommended Minimum Initialization Parameters

DB_NAME - CONTROL_FILES - MEMORY_TARGET

-- set parameter value

vahidusefzadeh@gmail.com

alter system set parameter=value scope=SPFILE|MEMORY|BOTH

-- delete a parameter from SPFILe

ALTER SYSTEM RESET log_archive_dest_4 SCOPE=BOTH SID='rac1';

--useful query:

--- undocumented parameters

SELECT a.ksppinm parameter, a.ksppdesc description, b.ksppstvl session_value, c.ksppstvl instance_value FROM x$ksppi a, x$ksppcv b, x$ksppsv c WHERE a.indx = b.indx AND a.indx = c.indx AND SUBSTR (a.ksppinm,1,1) = '_' ORDER BY a.ksppinm;

---display current value of a parameter

select * from v$parameter where name = 'spfile' ;

OR

show parameter db_name

---pfile or spfile?

SELECT DECODE(value, NULL, 'PFILE', 'SPFILE') "Init File Type" FROM sys.v_$parameter WHERE name = 'spfile';

---parameter alterd(last week)?

with h as (select distinct parameter_name,VALUE from dba_hist_parameter h, dba_hist_snapshot b WHERE h.snap_id = b.snap_id and b.begin_interval_time between sysdate-8 and sysdate -7 and (SELECT INSTANCE_NUMBER FROM V$INSTANCE)=h.instance_number) SELECT P.NAME,P.VALUE CURR_VALUE,H.VALUE HIST_VALUE FROM V$PARAMETER P,h where p.name=h.parameter_name and h.value!=p.value;

CONTROLFILE

--set parameter:

alter system set CONTROL_FILES ='+DATA01'

--Backing Up Control File

ALTER DATABASE BACKUP CONTROLFILE TO TRACE AS <file_name> REUSE <RESETLOGS | NORESETLOGS>;

ALTER DATABASE BACKUP CONTROLFILE TO TRACE;

ALTER DATABASE BACKUP CONTROLFILE TO trace as '/u01/usefdb/control01.sql';

ALTER DATABASE BACKUP CONTROLFILE TO TRACE noresetlogs;

ALTER DATABASE BACKUP CONTROLFILE TO TRACE resetlogs;

--CREATE CONTROLFILE

syntex

CREATE CONTROLFILE [REUSE] [SET] DATABASE database [logfile_clause] { RESETLOGS | NORESETLOGS }

[DATAFILE file_specification [, file_specification]...] [MAXLOGFILES integer | MAXLOGMEMBERS integer

| MAXLOGHISTORY integer | MAXDATAFILES integer | MAXINSTANCES integer

| { ARCHIVELOG | NOARCHIVELOG } | FORCE LOGGING]... [character_set_clause] ;

example

CREATE CONTROLFILE REUSE DATABASE "USEFDB" NORESETLOGS FORCE LOGGING ARCHIVELOG

MAXLOGFILES 192

MAXLOGMEMBERS 3

MAXDATAFILES 1024

MAXINSTANCES 32

MAXLOGHISTORY 140228

LOGFILE

GROUP 1 ('/u01/oracle/oradata/usefdb/redo01.rdo', '/u01/oracle/oradata/usefdb/redo/redo01.rdo') SIZE 500M BLOCKSIZE 512,

GROUP 2 ('/u01/oracle/oradata/usefdb/redo02.rdo', '/u01/oracle/oradata/usefdb/redo/redo02.rdo') SIZE 500M BLOCKSIZE 512,

GROUP 3 ('/u01/oracle/oradata/usefdb/redo03.rdo', '/u01/oracle/oradata/usefdb/redo/redo03.rdo') SIZE 500M BLOCKSIZE 512

DATAFILE

'/u01/oracle/oradata/usefdb/system01.dbf',

'/u01/oracle/oradata/usefdb/undotbs01.dbf',

'/u01/oracle/oradata/usefdb/sysaux01.dbf',

'/u01/oracle/oradata/usefdb/users01.dbf'

CHARACTER SET AR8MSWIN1256;

OR

vahidusefzadeh@gmail.com

CREATE CONTROLFILE REUSE DATABASE "USEFDB" RESETLOGS FORCE LOGGING ARCHIVELOG….

--Create Standby Control File

ALTER DATABASE CREATE LOGICAL STANDBY CONTROLFILE AS <file_name> REUSE;

ALTER DATABASE CREATE LOGICAL STANDBY CONTROLFILE AS 'c:\oragrid\control03.ctl';

ALTER DATABASE CREATE PHYSICAL STANDBY CONTROLFILE AS <file_name> REUSE;

ALTER DATABASE CREATE PHYSICAL STANDBY CONTROLFILE AS 'c:\oragrid\control03.ctl' REUSE;

--other related commands

recover database using backup controlfile;

RECOVER DATABASE USING BACKUP CONTROLFILE UNTIL CANCEL;

ALTER DATABASE RECOVER DATABASE UNTIL CANCEL USING BACKUP CONTROLFILE;

--Control File information

V$CONTROLFILE, v$controlfile_record_section

--useful query:

```
SELECT RPAD(SUBSTR(name, 1, 50), 51,' ') "CONTROL FILE NAME" FROM gv$controlfile;
```

Redo Log

-- Adding Online Redo Log File Groups

ALTER DATABASE ADD LOGFILE ('<log_member_path_and_name>', '<log_member_path_and_name>')
SIZE <integer> <K | M | G>;

ALTER DATABASE ADD LOGFILE ('/u01/usefdb/log1c.rdo', '/oracle/dbs/log2c.rdo') SIZE 500M;

-

ALTER DATABASE ADD LOGFILE GROUP <group_number> ('<log_member_path_and_name>') SIZE <integer> <K | M | G>;

ALTER DATABASE ADD LOGFILE GROUP 10 ('/u01/usefdb/log1c.rdo', '/oracle/dbs/log2c.rdo') SIZE 500M;

-

ALTER DATABASE ADD LOGFILE THREAD <integer> GROUP <integer> (<logfile_path_and_name>, <logfile_path_and_name>);

Alter database add logfile **thread** 2 group 8(/u01/usefdb/redo05.log);

- Adding Online Redo Log File Members

ALTER DATABASE ADD LOGFILE MEMBER '<log_member_path_and_name>' TO GROUP <group_number>;

ALTER DATABASE ADD LOGFILE MEMBER '/u01/usefdb /log3.rdo' TO GROUP 2;

-

ALTER DATABASE ADD LOGFILE GROUP <integer> (<logfile_path_and_name>, <logfile_path_and_name>) SIZE <integer><K | M | G>;

ALTER DATABASE ADD LOGFILE MEMBER '/oracle/dbs/log2c.rdo' TO ('/oracle/dbs/log2a.rdo', '/oracle/dbs/log2b.rdo');

--relocate redo log

ALTER DATABASE RENAME FILE '<existing_path_and_file_name>' TO '<new_path_and_file_name>';

ALTER DATABASE RENAME FILE '/diska/logs/log1a.rdo', '/diska/logs/log2a.rdo' TO '/diskc/logs/log1c.rdo', '/diskc/logs/log2c.rdo';

--Drop Online Redo Log File Groups

ALTER DATABASE DROP [STANDBY] LOGFILE GROUP <integer>;

ALTER DATABASE DROP LOGFILE GROUP 3;

-- Dropping Online Redo Log File Members

ALTER DATABASE DROP [STANDBY] LOGFILE MEMBER <logfile_member_path_and_name>;

ALTER DATABASE DROP LOGFILE MEMBER '/oracle/dbs/log3c.rdo';

-- Forcing a log switch :

ALTER SYSTEM SWITCH LOGFILE;

-- Clearing a Redo Log File:

ALTER DATABASE CLEAR LOGFILE GROUP 3;

ALTER DATABASE CLEAR UNARCHIVED LOGFILE GROUP 3;

--Dumping Log Files

ALTER SYSTEM DUMP LOGFILE '<logfile_path_and_name>' DBA MIN <file_number> <block_number> DBA MAX <file_number> <block_number>;

or

vahidusefzadeh@gmail.com

ALTER SYSTEM DUMP LOGFILE '<logfile_path_and_name>' TIME MIN <value> TIME MIN <value>;

ALTER SYSTEM DUMP LOGFILE 'c:\oracle\product\oradata\orabase\redo01.log' TIME MIN 4997217;

--view:

VLOG, VLOGFILE

--useful query:

SELECT * FROM V$LOG G, V$LOGFILE M where G.GROUP#=M.GROUP# order by M.GROUP#;

Archived Redo Log

-- Changing Database Archiving Mode

SHUTDOWN immediate

STARTUP MOUNT

ALTER DATABASE ARCHIVELOG;

ALTER DATABASE OPEN;

-- Performing Manual Archiving

ALTER DATABASE ARCHIVELOG MANUAL;

ALTER SYSTEM ARCHIVE LOG ALL;

-- Adjusting Number of Archiver Processes

ALTER SYSTEM SET LOG_ARCHIVE_MAX_PROCESSES=3;

-- Specifying Archive Destinations

LOG_ARCHIVE_DEST_1 = 'LOCATION = /disk1/archive'

LOG_ARCHIVE_DEST_4 = 'SERVICE = standby1'

--format archive log:

LOG_ARCHIVE_FORMAT=%t_%s_%r.dbf

-- Specifying Mandatory and Optional Destinations

LOG_ARCHIVE_MIN_SUCCEED_DEST

vahidusefzadeh@gmail.com

-- Controlling Archiving to a Destination

alter system set LOG_ARCHIVE_DEST_STATE_2 = DEFER

alter system set LOG_ARCHIVE_DEST_STATE_2 = ENABLE

-- Controlling Trace Output Generated by the Archivelog Process

ALTER SYSTEM SET LOG_ARCHIVE_TRACE=12;

--view:

V$ARCHIVE_DEST , V$ARCHIVED_LOG, V$ARCHIVE_DEST, V$ARCHIVE_PROCESSES,V$BACKUP_REDOLOG, V$LOG, V$LOG_HISTORY

--useful query:

select LOG_MODE from V$DATABASE;

--count arcs per day

```
SELECT  trunc(first_time) "Date",
       to_char(first_time, 'Dy') "Day",
       count(1) "Total",
       SUM(decode(to_char(first_time, 'hh24'),'00',1,0)) "h0",
       SUM(decode(to_char(first_time, 'hh24'),'01',1,0)) "h1",
       SUM(decode(to_char(first_time, 'hh24'),'02',1,0)) "h2",
       SUM(decode(to_char(first_time, 'hh24'),'03',1,0)) "h3",
       SUM(decode(to_char(first_time, 'hh24'),'04',1,0)) "h4",
       SUM(decode(to_char(first_time, 'hh24'),'05',1,0)) "h5",
       SUM(decode(to_char(first_time, 'hh24'),'06',1,0)) "h6",
       SUM(decode(to_char(first_time, 'hh24'),'07',1,0)) "h7",
       SUM(decode(to_char(first_time, 'hh24'),'08',1,0)) "h8",
       SUM(decode(to_char(first_time, 'hh24'),'09',1,0)) "h9",
       SUM(decode(to_char(first_time, 'hh24'),'10',1,0)) "h10",
       SUM(decode(to_char(first_time, 'hh24'),'11',1,0)) "h11",
       SUM(decode(to_char(first_time, 'hh24'),'12',1,0)) "h12",
       SUM(decode(to_char(first_time, 'hh24'),'13',1,0)) "h13",
       SUM(decode(to_char(first_time, 'hh24'),'14',1,0)) "h14",
       SUM(decode(to_char(first_time, 'hh24'),'15',1,0)) "h15",
       SUM(decode(to_char(first_time, 'hh24'),'16',1,0)) "h16",
       SUM(decode(to_char(first_time, 'hh24'),'17',1,0)) "h17",
```

```
    SUM(decode(to_char(first_time, 'hh24'),'18',1,0)) "h18",
    SUM(decode(to_char(first_time, 'hh24'),'19',1,0)) "h19",
    SUM(decode(to_char(first_time, 'hh24'),'20',1,0)) "h20",
    SUM(decode(to_char(first_time, 'hh24'),'21',1,0)) "h21",
    SUM(decode(to_char(first_time, 'hh24'),'22',1,0)) "h22",
    SUM(decode(to_char(first_time, 'hh24'),'23',1,0)) "h23",
    -- for today, use # of hrs so far today to get avg
    -- for past days, use 24 hrs to get avg
    decode(trunc(first_time),
        trunc(sysdate), round(count(1) / (24 * to_number(to_char(sysdate, 'sssss')+1) / 86400),2),
        round(count(1) / 24, 2)) "Avg"
from GV$log_history
where 1=1
group by trunc(first_time), to_char(first_time, 'Dy')
Order by 1;
```

--- max sequence#

```
SELECT INCARNATION#, RESETLOGS_ID, STATUS FROM V$DATABASE_INCARNATION ;
select max(sequence#),thread# from v$archived_log where THREAD#=1 and RESETLOGS_ID=865773832 group by thread#;
```

Tablespace

--CREATE TABLESPACE

CREATE [BIGFILE | SMALLFILE] { permanent_tablespace_clause | temporary_tablespace_clause | undo_tablespace_clause } ;

example

-- Creating a Locally Managed Tablespace

CREATE TABLESPACE useftbs DATAFILE '/u01/oracle/usefdb/useftbs01.dbf' SIZE 50M

EXTENT MANAGEMENT AUTOALLOCATE;

CREATE TABLESPACE useftbs DATAFILE '/u01/oracle/usefdb/useftbs01.dbf' SIZE 50M

EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K; -- default 1MB

--- creating a tablespace in an ASM diskgroup

CREATE TABLESPACE sample DATAFILE '+dgroup1';

CREATE TABLESPACE satbs DATAFILE '+DATA' size 50m;

--ALTER TABLESPACE

syntax

ALTER TABLESPACE tablespace { DEFAULT [table_compression] storage_clause

| MINIMUM EXTENT size_clause | RESIZE size_clause | COALESCE | SHRINK SPACE [KEEP size_clause]

| RENAME TO new_tablespace_name | { BEGIN | END } BACKUP | datafile_tempfile_clauses

| tablespace_logging_clauses | tablespace_group_clause | tablespace_state_clauses

| autoextend_clause | flashback_mode_clause | tablespace_retention_clause } ;

-- ADD DATAFILE

ALTER TABLESPACE useftbs ADD DATAFILE '/u01/oracle/usefdb/useftbs02.dbf' SIZE 1M;

-- Bigfile Tablespaces

CREATE BIGFILE TABLESPACE bigtbs DATAFILE '/u01/oracle/usefdb/bigtbs01.dbf' SIZE 50G

--Altering a Bigfile Tablespace

ALTER TABLESPACE bigtbs RESIZE 80G;

ALTER TABLESPACE bigtbs AUTOEXTEND ON NEXT 20G;

-- Creating a Locally Managed Temporary Tablespace

CREATE TEMPORARY TABLESPACE useftemp TEMPFILE '/u01/oracle/usefdb/useftemp01.dbf'

SIZE 20M REUSE EXTENT MANAGEMENT LOCAL UNIFORM SIZE 16M;

-- Altering a Locally Managed Temporary Tablespace

ALTER TABLESPACE useftemp ADD TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf' SIZE 18M REUSE;

ALTER TABLESPACE useftemp TEMPFILE OFFLINE;

ALTER TABLESPACE useftemp TEMPFILE ONLINE;

ALTER DATABASE TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf' OFFLINE;

ALTER DATABASE TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf' ONLINE;

ALTER DATABASE TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf' RESIZE 18M;

ALTER DATABASE TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf' DROP INCLUDING DATAFILES;

-- Shrinking a Locally Managed Temporary Tablespace

ALTER TABLESPACE useftemp1 SHRINK SPACE KEEP 20M;

ALTER TABLESPACE useftemp2 SHRINK TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf';

-- Default Temporary Tablespace

SELECT PROPERTY_NAME, PROPERTY_VALUE FROM DATABASE_PROPERTIES WHERE PROPERTY_NAME='DEFAULT_TEMP_TABLESPACE';

-- Tablespace Groups

CREATE TEMPORARY TABLESPACE <tablespace_name> TEMP <data_file_path_and_name> SIZE <integer> <K | M | G | T | P | E> TABLESPACE GROUP <group_name>;

CREATE TEMPORARY TABLESPACE useftemp2 TEMPFILE '/u01/oracle/usefdb/useftemp201.dbf' SIZE 50M TABLESPACE GROUP group1;

-

ALTER TABLESPACE <tablespace_name> TABLESPACE GROUP <group_name>;

ALTER TABLESPACE useftemp TABLESPACE GROUP group2;

-

ALTER DATABASE <tablespace_name> DEFAULT TEMPORARY TABLESPACE <group_name>;

CREATE TEMPORARY TABLESPACE useftemp3 TEMPFILE '/u01/oracle/usefdb/useftemp301.dbf' SIZE 25M TABLESPACE GROUP group1;

-

ALTER TABLESPACE useftemp2 TABLESPACE GROUP group2;

ALTER TABLESPACE useftemp3 TABLESPACE GROUP '';

ALTER DATABASE sample DEFAULT TEMPORARY TABLESPACE group2;

vahidusefzadeh@gmail.com

-- Compressed Tablespaces

CREATE TABLESPACE ... DEFAULT ROW STORE COMPRESS ADVANCED ... ;

-- Encrypted Tablespaces

CREATE TABLESPACE securespace DATAFILE '/u01/oracle/oradata/usefdb/ secure01.dbf' SIZE 100M

ENCRYPTION DEFAULT STORAGE(ENCRYPT);

CREATE TABLESPACE securespace DATAFILE '/u01/oracle/oradata/usefdb/ secure01.dbf' SIZE 100M ENCRYPTION USING 'AES256' DEFAULT STORAGE(ENCRYPT);

SELECT t.name, e.encryptionalg algorithm FROM v$tablespace t, v$encrypted_tablespaces e

WHERE t.ts# = e.ts#;

-- Controlling Tablespaces Availability

ALTER TABLESPACE users OFFLINE NORMAL;

ALTER TABLESPACE users ONLINE;

ALTER TABLESPACE flights READ ONLY;

ALTER TABLESPACE flights READ WRITE;

--Renaming Tablespaces

ALTER TABLESPACE users RENAME TO usersts;

-- DROP TABLESPACE

syntax

DROP TABLESPACE tablespace [INCLUDING CONTENTS [{AND | KEEP} DATAFILES][CASCADE CONSTRAINTS]] ;

examples

DROP TABLESPACE users INCLUDING CONTENTS;

DROP TABLESPACE test01 CASCADE CONSTRAINTS;

DROP TABLESPACE users INCLUDING CONTENTS CASCADE CONSTRAINTS;

DROP TABLESPACE users INCLUDING CONTENTS AND DATAFILES;

vahidusefzadeh@gmail.com

--tablespace free space

```
select TABLESPACE_NAME,round(sum(bytes/1000000000)) "GB"  from  dba_free_space  group by
TABLESPACE_NAME;
```

-- Default Permanent Tabelspace

```
SELECT property_value FROM database_properties WHERE  property_name=
'DEFAULT_PERMANENT_TABLESPACE';

ALTER DATABASE DEFAULT TABLESPACE users;
```

--view

```
DBA_FREE_SPACE, DBA_DATA_FILES, DBA_TABLESPACES

select * from dba_tablespace_usage_metrics;
```

--useful query:

```
select * from dba_tablespace_usage_metrics;
```

--- tablespace size usage report

```
SELECT df.tablespace_name tbspname,  sum(df.bytes)/1024/1024 tbsp_size,  nvl(sum(e.used_bytes)/1024/1024,0)
used,  nvl(sum(f.free_bytes)/1024/1024,0) free,  nvl((sum(e.used_bytes)*100)/sum(df.bytes),0) pct_used
FROM DBA_DATA_FILES df,  (SELECT file_id,   SUM(nvl(bytes,0)) used_bytes   FROM dba_extents
  GROUP BY file_id) e,  (SELECT MAX(bytes) free_bytes, file_id   FROM dba_free_space
  GROUP BY file_id) f WHERE e.file_id(+) = df.file_id   AND df.file_id = f.file_id(+)   GROUP BY df.tablespace_name
ORDER BY 5 DESC
```

--- size usage report for a specific tablespace

```
SELECT df.tablespace_name tbspname,  sum(df.bytes)/1024/1024 tbsp_size,
 nvl(sum(e.used_bytes)/1024/1024,0) used,  nvl(sum(f.free_bytes)/1024/1024,0) free,
nvl((sum(e.used_bytes)*100)/sum(df.bytes),0) pct_use FROM DBA_DATA_FILES df,  (SELECT file_id,
SUM(nvl(bytes,0)) used_bytes   FROM dba_extents   WHERE TABLESPACE_NAME='USERS'   GROUP BY file_id) e,
(SELECT MAX(bytes) free_bytes, file_id   FROM dba_free_space   WHERE TABLESPACE_NAME='USERS'   GROUP BY
file_id) f WHERE e.file_id(+) = df.file_id   AND df.file_id = f.file_id(+)   AND TABLESPACE_NAME='USERS'
  GROUP BY df.tablespace_name;
```

---tablespace sizes

```
SELECT DF.TABLESPACE_NAME TBSPNAME, ROUND(SUM(DF.BYTES)/1024/1024/1024,2) GB,   COUNT(FILE_NAME)
DATAFILES FROM DBA_DATA_FILES DF  GROUP BY ROLLUP(DF.TABLESPACE_NAME )  ORDER BY 1
```

---free space in temp tablespaces:

select sum(free_blocks) from gv$sort_segment where tablespace_name = 'TEMP' ;

---tablespace info

select TS#,NAME,INCLUDED_IN_DATABASE_BACKUP, BIGFILE,FLASHBACK_ON,ENCRYPT_IN_BACKUP from V$TABLESPACE;

--- tablespace groups

select GROUP_NAME, TABLESPACE_NAME from DBA_TABLESPACE_GROUPS order BY TABLESPACE_NAME

Data File

- Creating Datafile

syntax

CREATE TABLESPACE ….

CREATE TEMPORARY TABLESPACE ….

CREATE DATABASE …

ALTER TABLESPACE ADD TEMPFILE <temp_file_name>;

ALTER TABLESPACE <tablespace_name> ADD DATAFILE '<path_and_file_name>' SIZE <n>K | M | G | T | P | E;

ALTER DATABASE CREATE DATAFILE '<path_and_file_name>' SIZE <integer><K | M | G | T | P | E> AS '<tablespace_name>';

----example:

ALTER TABLESPACE users ADD DATAFILE '/u01/oracle/usefdb/users03.dbf' SIZE 10M AUTOEXTEND ON NEXT 512K MAXSIZE 250M;

--Enabling and Disabling Automatic Extension

ALTER DATABASE DATAFILE <'file_path_and_name' | file_number> AUTOEXTEND <OFF | ON [NEXT SIZE <integer><K | M | G | T | P | E> MAXSIZE <UNLIMITED | <integer><K | M | G | T | P | E>;

ALTER TABLESPACE users ADD DATAFILE '/u01/oracle/usefdb/users03.dbf' SIZE 10M AUTOEXTEND ON NEXT 512K MAXSIZE 250M;

-- Resizing a Datafile

ALTER DATABASE DATAFILE <'file_path_and_name' | file_number> RESIZE <integer><K | M | G | T | P | E>;

ALTER DATABASE DATAFILE '/u01/oracle/usefdb/stuff01.dbf'   RESIZE 100M;

-- END BACKUP

ALTER DATABASE DATAFILE <'file_path_and_name' | file_number> END BACKUP;

ALTER DATABASE DATAFILE '/app/oracle/oradata/uefdb/users01.dbf' END BACKUP;

--Altering Data File Availability

ALTER DATABASE DATAFILE <'file_path_and_name' | file_number> <ONLINE | OFFLINE [FOR DROP]>;

ALTER DATABASE DATAFILE '/u01/oracle/usefdb/stuff01.dbf' ONLINE;

ALTER DATABASE DATAFILE '/u01/oracle/usefdb/stuff01.dbf' OFFLINE;

ALTER TABLESPACE ... DATAFILE {ONLINE|OFFLINE}

ALTER TABLESPACE ... TEMPFILE {ONLINE|OFFLINE}

--Taking Data Files Offline in NOARCHIVELOG Mode

ALTER DATABASE DATAFILE '/u01/oracle/usefdb/users03.dbf' OFFLINE FOR DROP;

--Rename File

ALTER DATABASE RENAME FILE '<current_file_name>' TO '<new_file_name>'

ALTER DATABASE RENAME FILE '/u01/oradata/tools01.dbf' TO '/u06/oradata/tools01.dbf';

-- Relocating an Online Data File

ALTER DATABASE MOVE DATAFILE <['path_and_file_name' | 'ASM_file_name' | file_number]>
TO <['path_and_file_name' | 'ASM_file_name'> [REUSE] [KEEP];

ALTER DATABASE MOVE DATAFILE '/u01/oracle/usefdb/user1.dbf'   TO '/u02/oracle/usefdb/user01.dbf';

--Copying an Online Data File

ALTER DATABASE MOVE DATAFILE '/u01/oracle/usefdb/user1.dbf'   TO '/u02/oracle/usefdb/user1.dbf' KEEP;

-- Relocating an Online Data File and Overwriting an Existing File

ALTER DATABASE MOVE DATAFILE '/u01/oracle/usefdb/user1.dbf'   TO '/u02/oracle/usefdb/user1.dbf' REUSE;

-- Relocating an Online Data File to Oracle ASM

ALTER DATABASE MOVE DATAFILE '/u01/oracle/usefdb/user1.dbf' TO '+dg01/data/usefdb/datafile/user1.dbf';

--Moving a File from One ASM Location to Another ASM Location

ALTER DATABASE MOVE DATAFILE '+dg01/data/usefdb/datafile/user1.dbf' TO '+dg02/data/usefdb/datafile/user1.dbf';

-- Renaming and Relocating Offline Data Files

ALTER TABLESPACE users OFFLINE NORMAL;

ALTER TABLESPACE users RENAME DATAFILE '/u01/oracle/usefdb/user1.dbf', '/u01/oracle/usefdb/user2.dbf' TO '/u02/oracle/usefdb/users01.dbf', '/u02/oracle/usefdb/users02.dbf';

ALTER TABLESPACE users ONLINE

SELECT FILE_NAME, BYTES FROM DBA_DATA_FILES WHERE TABLESPACE_NAME = 'USERS';

--Dropping Data Files

ALTER DATABASE DATAFILE '<file_name_or_file_number>' [offline] DROP;

ALTER TABLESPACE example DROP DATAFILE '+DGROUP1/example_df3.f';

--drop tempfile

ALTER TABLESPACE useftemp DROP TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf';

ALTER DATABASE TEMPFILE '/u01/oracle/usefdb/useftemp02.dbf' DROP INCLUDING DATAFILES;

-- View

DBA_DATA_FILES ,DBA_EXTENTS , USER_EXTENTS , DBA_FREE_SPACE , USER_FREE_SPACE , V$DATAFILE , V$DATAFILE_HEADER

--useful query:

--data file info:

SELECT FILE#,T.NAME TABLESPACE_NAME,D.NAME FILENAME, CREATION_CHANGE#,CREATION_TIME, RFILE#,STATUS, ENABLED,CHECKPOINT_CHANGE# ,CHECKPOINT_TIME, UNRECOVERABLE_CHANGE#,UNRECOVERABLE_TIME,LAST_CHANGE#, LAST_TIME, OFFLINE_CHANGE#,ONLINE_CHANGE#, ONLINE_TIME,BYTES/1024/1024 FILESIZE_MB,BLOCKS, CREATE_BYTES,BLOCK_SIZE, PLUGGED_IN,BLOCK1_OFFSET, AUX_NAME FROM V$DATAFILE D, V$TABLESPACE T WHERE D.TS# = T.TS# ORDER BY TABLESPACE_NAME, D.RFILE#;

---

vahidusefzadeh@gmail.com

Undo

--parameters:

UNDO_TABLESPACE = undotbs_01

UNDO_MANAGEMENT= AUTO or MANUAL

UNDO_RETENTION=number(second)

--Enabling Automatic Undo Management

select value from v$parameter where upper(name)='UNDO_MANAGEMENT';

alter system set UNDO_MANAGEMENT=AUTO scope=spfile ;

--Creating an Undo Tablespace

CREATE DATABASE usefdb CONTROLFILE REUSE UNDO TABLESPACE undotbs_01 DATAFILE '/u01/oracle/usefdb/undo0101.dbf';

CREATE UNDO TABLESPACE undotbs_02 DATAFILE '/u01/oracle/usefdb/undo0201.dbf' SIZE 2M REUSE AUTOEXTEND ON;

-- Fixed-Size Undo Tablespace

ALTER DATABASE DATAFILE '/oracle/dbs/undotbs.dbf' AUTOEXTEND OFF;

--Altering an Undo Tablespace

ALTER TABLESPACE undotbs_01 ADD DATAFILE '/u01/oracle/usefdb/undo0102.dbf' AUTOEXTEND ON NEXT 1M MAXSIZE UNLIMITED;

--Dropping an Undo Tablespace

DROP TABLESPACE undotbs_01;

--Switching Undo Tablespaces

ALTER SYSTEM SET UNDO_TABLESPACE = undotbs_02;

--Enabling and Disabling Temporary Undo

ALTER SESSION SET TEMP_UNDO_ENABLED = TRUE;

ALTER SESSION SET TEMP_UNDO_ENABLED = FALSE;

--Enabling Retention Guarantee

vahidusefzadeh@gmail.com

select RETENTION from DBA_TABLESPACES where TABLESPACE_NAME='UNDOTBS1';

create undo tablespace undotbs01 .. RETENTION GUARANTEE;

alter tablespace undotbs1 RETENTION GUARANTEE;

alter tablespace undotbs1 RETENTION NOGUARANTEE;

--view

v$UNDOSTAT

--useful query:

---undo sizes by STATUS

select e.TABLESPACE_NAME, e.STATUS, to_char(sum(e.BYTES/1024),'999,999,999,999') SIZE_KB from DBA_UNDO_EXTENTS e group by e.TABLESPACE_NAME, e.STATUS order by e.STATUS

---undo sizes consumed by active transactions

SELECT s.username, sum(t.used_ublk) used_undo_blocks from v$session s, v$transaction t where s.saddr = t.ses_addr and t.status='ACTIVE' group by s.username order by s.username;

---maximum query time

SELECT round(MAX(maxquerylen)/60) Minutes FROM v$undostat;

---SNAPSHOT TOO OLD:

select 'Number of "ORA-01555 (Snapshot too old)" encountered since the last startup of the instance : ' || sum(ssolderrcnt) from v$undostat;

---

objects

Table

--Creating a Table

syntax

CREATE [GLOBAL TEMPORARY] TABLE [schema.] table { relational_table | object_table | XMLType_table }

example

CREATE TABLE usef.admin_emp (empno NUMBER(5) PRIMARY KEY, ename VARCHAR2(15) NOT NULL, ssn

NUMBER(9) ENCRYPT USING 'AES256', job VARCHAR2(10), mgr NUMBER(5), hiredate DATE DEFAULT (sysdate),

photo BLOB, sal NUMBER(7,2), hrly_rate NUMBER(7,2) GENERATED ALWAYS AS (sal/2080), deptno NUMBER(3)

NOT NULL CONSTRAINT admin_dept_fkey REFERENCES usef.departments (department_id), comments

VARCHAR2(32767), status VARCHAR2(10) INVISIBLE) TABLESPACE admin_tbs STORAGE (INITIAL 50K);

-- creates a temporary table

CREATE GLOBAL TEMPORARY TABLE admin_work_area (startdate DATE, enddate DATE, class CHAR(20))

ON COMMIT DELETE ROWS;

-- Parallelizing Table Creation

CREATE TABLE usef.admin_emp_dept PARALLEL COMPRESS AS SELECT * FROM usef.employees WHERE department_id = 10;

-- Alter … parallel

Alter table a parallel 5;

--Creating a Table with Advanced Row Compression

CREATE TABLE orders ... ROW STORE COMPRESS ADVANCED;

--Creating a Table with Basic Table Compression

CREATE TABLE sales_history ... ROW STORE COMPRESS BASIC;

CREATE TABLE sales_history ... ROW STORE COMPRESS;

--Using Direct-Path Insert to Insert Rows Into a Table

INSERT /*+ APPEND */ INTO sales_history SELECT * FROM sales WHERE cust_id=8890;

COMMIT;

-- Creating a Table with Warehouse Compression

CREATE TABLE sales_history ... COLUMN STORE COMPRESS FOR QUERY;

-- COMMENT

syntax

COMMENT ON { AUDIT POLICY policy | COLUMN [schema.] { table. | view. | materialized_view. } column

| EDITION edition_name | INDEXTYPE [schema.] indextype | MATERIALIZED VIEW materialized_view

| MINING MODEL [schema.] model | OPERATOR [schema.] operator | TABLE [schema.] { table | view }

} IS string ;

example

COMMENT ON TABLE usef.admin_emp IS 'Enhanced employee table';

-- Determining If a Table Is Compressed

SELECT table_name, compression, compress_for FROM user_tables;

SELECT table_name, partition_name, compression, compress_for FROM user_tab_partitions;

-- Determining Which Rows Are Compressed

SELECT DECODE(DBMS_COMPRESSION.GET_COMPRESSION_TYPE(ownname => 'HR',

objname => 'EMPLOYEES', subobjname => '', row_id => 'AAAVEIAAGAAAABTAAD'), 1, 'No Compression', 2,

'Advanced Row Compression', 4, 'Hybrid Columnar Compression for Query High', 8, 'Hybrid Columnar

Compression for Query Low', 16, 'Hybrid Columnar Compression for Archive High', 32, 'Hybrid Columnar

Compression for Archive Low', 4096, 'Basic Table Compression', 'Unknown Compression Type')

compression_type FROM DUAL;

--ALTER TABLE

syntax

ALTER TABLE [schema.] table [alter_table_properties | column_clauses | constraint_clauses

| alter_table_partitioning | alter_external_table | move_table_clause | modify_opaque_type

] [enable_disable_clause | { ENABLE | DISABLE } { TABLE LOCK | ALL TRIGGERS }] ... ;

--Changing Compression Level

ALTER TABLE ... MOVE PARTITION ... ONLINE

ALTER TABLE ... MOVE SUBPARTITION ... ONLINE

-- Invisible Columns

CREATE TABLE mytable (a INT, b INT INVISIBLE, c INT);

ALTER TABLE mytable MODIFY (b VISIBLE);

ALTER TABLE mytable2 MODIFY (y VISIBLE);

vahidusefzadeh@gmail.com

-- Moving a Table to a New Segment or Tablespace

ALTER TABLE ... MOVE PARTITION ... ONLINE

ALTER TABLE ... MOVE SUBPARTITION ... ONLINE

-- Moving a Table

ALTER TABLE usef.jobs MOVE STORAGE (INITIAL 20K NEXT 40K MINEXTENTS 2 MAXEXTENTS 20 PCTINCREASE 0) TABLESPACE usef_tbs;

--Moving a Table Partition to a New Segment

ALTER TABLE sales MOVE PARTITION sales_q4_2003 ROW STORE COMPRESS ADVANCED UPDATE INDEXES ONLINE;

--Modifying an Existing Column Definition

ALTER TABLE oe.product_information MODIFY(product_description VARCHAR2(32767));

----Adding Table Columns

ALTER TABLE usef.admin_emp ADD (bonus NUMBER (7,2));

--Renaming Table Columns

ALTER TABLE usef.admin_emp RENAME COLUMN comm TO commission;

--Dropping Table Columns

ALTER TABLE usef.admin_emp DROP COLUMN sal;

ALTER TABLE usef.admin_emp DROP (bonus, commission);

--Marking Columns Unused

ALTER TABLE usef.admin_emp SET UNUSED (hiredate, mgr);

--Removing Unused Columns

ALTER TABLE usef.admin_emp DROP UNUSED COLUMNS CHECKPOINT 250;

--Placing a Table in Read-Only Mode

ALTER TABLE SALES READ ONLY;

ALTER TABLE SALES READ WRITE;

--All DML operations on the table or any of its partitions

TRUNCATE TABLE

SELECT FOR UPDATE

ALTER TABLE ADD/MODIFY/RENAME/DROP COLUMN

ALTER TABLE SET COLUMN UNUSED

ALTER TABLE DROP/TRUNCATE/EXCHANGE (SUB)PARTITION

ALTER TABLE UPGRADE INCLUDING DATA or ALTER TYPE CASCADE INCLUDING TABLE

Online redefinition

FLASHBACK TABLE

--The following operations are permitted on a read-only table:

SELECT

CREATE/ALTER/DROP INDEX

ALTER TABLE ADD/MODIFY/DROP/ENABLE/DISABLE CONSTRAINT

ALTER TABLE for physical property changes

ALTER TABLE DROP UNUSED COLUMNS

ALTER TABLE ADD/COALESCE/MERGE/MODIFY/MOVE/RENAME/SPLIT (SUB)PARTITION

ALTER TABLE MOVE

ALTER TABLE ENABLE ROW MOVEMENT and ALTER TABLE SHRINK

RENAME TABLE and ALTER TABLE RENAME TO

DROP TABLE

ALTER TABLE DEALLOCATE UNUSED

ALTER TABLE ADD/DROP SUPPLEMENTAL LOG

--DROP TABLE

syntax

DROP TABLE [schema.] table [CASCADE CONSTRAINTS] [PURGE] ;

example

DROP TABLE usef.int_admin_emp;

DROP TABLE usef.admin_emp CASCADE CONSTRAINTS;

DROP TABLE usef.admin_emp PURGE;

--PURGE

syntax

PURGE { { TABLE table | INDEX index } | { RECYCLEBIN | DBA_RECYCLEBIN } | TABLESPACE tablespace [USER username] } ;

--Truncating Tables or Clusters

syntax

--TRUNCATE_CLUSTER

TRUNCATE CLUSTER [schema.] cluster [{DROP | REUSE} STORAGE] ;

--TRUNCATE_TABLE

TRUNCATE TABLE [schema.] table [{PRESERVE | PURGE} MATERIALIZED VIEW LOG] [{DROP [ALL] | REUSE} STORAGE] [CASCADE] ;

example

TRUNCATE TABLE emp_dept DROP STORAGE;

TRUNCATE TABLE emp_dept REUSE STORAGE;

TRUNCATE TABLE emp DROP ALL STORAGE;

--Creating Index-Organized Tables

CREATE TABLE admin_docindex(token char(20), doc_id NUMBER, token_frequency NUMBER, token_offsets

VARCHAR2(2000), CONSTRAINT pk_admin_docindex PRIMARY KEY (token, doc_id)) ORGANIZATION INDEX

TABLESPACE admin_tbs PCTTHRESHOLD 20 OVERFLOW TABLESPACE admin_tbs2;

--Parallelizing Index-Organized Table Creation

CREATE TABLE admin_iot3(i PRIMARY KEY, j, k, l) ORGANIZATION INDEX PARALLEL AS SELECT * FROM usef.jobs;

--Using Key Compression

CREATE TABLE admin_iot5(i INT, j INT, k INT, l INT, PRIMARY KEY (i, j, k)) ORGANIZATION INDEX COMPRESS;

CREATE TABLE admin_iot6(i INT, j INT, k INT, l INT, PRIMARY KEY(i, j, k)) ORGANIZATION INDEX COMPRESS 2;

```
CREATE TABLE admin_iot7(i INT, j INT, k INT, l INT, PRIMARY KEY (i, j, k))   ORGANIZATION INDEX COMPRESS 1;

ALTER TABLE admin_iot5 MOVE NOCOMPRESS;
```

--Altering Index-Organized Tables

```
ALTER TABLE admin_docindex INITRANS 4 OVERFLOW INITRANS 6;

ALTER TABLE admin_docindex PCTTHRESHOLD 15 INCLUDING doc_id;

ALTER TABLE admin_iot3 ADD OVERFLOW TABLESPACE admin_tbs2;
```

--Moving (Rebuilding) Index-Organized Tables

```
ALTER TABLE admin_docindex MOVE;

ALTER TABLE admin_docindex MOVE ONLINE;

ALTER TABLE admin_docindex MOVE TABLESPACE admin_tbs2  OVERFLOW TABLESPACE admin_tbs3;

ALTER TABLE admin_iot_lob MOVE LOB (admin_lob) STORE AS (TABLESPACE admin_tbs3);
```

--Creating Secondary Indexes on Index-Organized Tables

```
CREATE INDEX Doc_id_index on Docindex(Doc_id, Token);

SELECT Token FROM Docindex WHERE Doc_id = 1;
```

--Analyzing Index-Organized Tables

```
EXECUTE DBMS_STATS.GATHER_TABLE_STATS ('HR','COUNTRIES');
```

--Validating Tables, Indexes, Clusters, and Materialized Views

---ANALYZE

syntax

```
ANALYZE  { { TABLE [ schema. ] table    | INDEX [ schema. ] index    } [ partition_extension_clause ]
  | CLUSTER [ schema. ] cluster  } { validation_clauses  | LIST CHAINED ROWS [ into_clause ]
  | DELETE [ SYSTEM ] STATISTICS  } ;
```

example

```
ANALYZE TABLE emp VALIDATE STRUCTURE;

ANALYZE TABLE emp VALIDATE STRUCTURE CASCADE;
```

```
ANALYZE TABLE emp VALIDATE STRUCTURE CASCADE ONLINE;

ANALYZE INDEX loc_country_ix VALIDATE STRUCTURE;
```

--Modifying, Renaming, or Dropping Existing Integrity Constraints

```
ALTER TABLE dept DISABLE CONSTRAINT dname_ukey;

ALTER TABLE dept DISABLE PRIMARY KEY KEEP INDEX, DISABLE UNIQUE (dname, loc)

KEEP INDEX;

ALTER TABLE dept DISABLE PRIMARY KEY CASCADE;

ALTER TABLE dept ENABLE NOVALIDATE CONSTRAINT dname_ukey;

ALTER TABLE dept ENABLE NOVALIDATE PRIMARY KEY, ENABLE NOVALIDATE UNIQUE

(dname, loc);

ALTER TABLE dept MODIFY CONSTRAINT dname_key VALIDATE;

ALTER TABLE dept MODIFY PRIMARY KEY ENABLE NOVALIDATE;

 ALTER TABLE dept RENAME CONSTRAINT dname_ukey TO dname_unikey;

ALTER TABLE dept DROP UNIQUE (dname, loc);

ALTER TABLE emp  DROP PRIMARY KEY KEEP INDEX, DROP CONSTRAINT dept_fkey;
```

--Renaming Schema Objects

```
rename mytable to mytable2
```

--gathering statistics

```
Exec DBMS_STATS.GATHER_SCHEMA_STATS(ownname =>'SCOTT',estimate_percent=>10, degree=>1,
cascade=>TRUE, options=>'GATHER STALE');

EXEC DBMS_STATS.GATHER_SCHEMA_STATS('SCOTT');

EXEC DBMS_STATS.GATHER_SCHEMA_STATS(OWNNAME=>'MRT');

EXEC DBMS_STATS.GATHER_SCHEMA_STATS('SCOTT',ESTIMATE_PERCENT=>10);

EXEC DBMS_STATS.GATHER_TABLE_STATS('SCOTT','EMP');

EXEC DBMS_STATS.GATHER_TABLE_STATS('SCOTT','EMP',ESTIMATE_PERCENT=>15);

EXEC DBMS_STATS.GATHER_INDEX_STATS('SCOTT','EMP_PK');

EXEC DBMS_STATS.GATHER_INDEX_STATS('SCOTT','EMP_PK',ESTIMATE_PERCENT=>15);
```

vahidusefzadeh@gmail.com

--delete statistics

```
EXEC DBMS_STATS.DELETE_DATABASE_STATS;

EXEC DBMS_STATS.DELETE_SCHEMA_STATS('SCOTT');

EXEC DBMS_STATS.DELETE_TABLE_STATS('SCOTT','EMP');

EXEC DBMS_STATS.DELETE_INDEX_STATS('SCOTT','EMP_PK');

EXEC DBMS_STATS.DELETE_PENDING_STATS('SH','SALES');

 EXEC DBMS_STATS.GATHER_SCHEMA_STATS(OWNNAME=>'"DWH"',OPTIONS=>'GATHER  AUTO');

EXEC DBMS_STATS.GATHER_SCHEMA_STATS(OWNNAME=>'PERFSTAT',CASCADE=>TRUE);
```

--Switching to a Different Schema

```
ALTER SESSION SET CURRENT_SCHEMA = joe;
```

--Shrinking Database Segments Online

```
ALTER TABLE names  ENABLE ROW MOVEMENT -- it acquires table lock

ALTER TABLE names SHRINK SPACE; -- it acquires table lock

ALTER TABLE names SHRINK SPACE COMPACT; -- no table lock

ALTER TABLE names SHRINK SPACE CASCADE;

ALTER TABLE names MODIFY LOB (perf_review) (SHRINK SPACE);

ALTER TABLE names MODIFY PARTITION cust_P1 SHRINK SPACE;

ALTER TABLE names SHRINK SPACE CASCADE;

ALTER TABLE names OVERFLOW SHRINK SPACE;
```

-- Deallocating Unused Space

```
ALTER TABLE mytable DEALLOCATE UNUSED KEEP integer;

ALTER INDEX myindex DEALLOCATE UNUSED KEEP integer;

ALTER CLUSTER cluster DEALLOCATE UNUSED KEEP integer;
```

--useful query

```
select table_name from all_tables where table_name like '%TABLE%';
```

vahidusefzadeh@gmail.com

TRIGGER, PROCEDURE, PACKAGE, FUNCTION

--CREATE TRIGGER

syntax

CREATE [OR REPLACE][EDITIONABLE | NONEDITIONABLE]TRIGGER plsql_trigger_source

example

```
CREATE trigger FOR dml_event_clause ON view
COMPOUND TRIGGER
INSTEAD OF EACH ROW IS BEGIN
 statement;
END INSTEAD OF EACH ROW;
/
```

--ALTER TRIGGER

syntax

ALTER TRIGGER [schema.] trigger_name { trigger_compile_clause | { ENABLE| DISABLE } | RENAME TO new_name | { EDITIONABLE | NONEDITIONABLE } } ;

example

```
ALTER TRIGGER update_job_history DISABLE;
ALTER TRIGGER update_job_history ENABLE;
```

--Enabling and Disabling Triggers

```
select table_name, trigger_name from user_triggers where status='DISABLED';
ALTER TRIGGER reorder ENABLE;
ALTER TRIGGER reorder DISABLE;
ALTER TABLE inventory ENABLE ALL TRIGGERS;
ALTER TABLE inventory DISABLE ALL TRIGGERS;
```

--DROP TRIGGER

DROP TRIGGER [schema.] trigger ;

--CREATE PROCEDURE

syntax

CREATE [OR REPLACE][EDITIONABLE | NONEDITIONABLE] PROCEDURE plsql_procedure_source

example

```
CREATE PROCEDURE remove_emp (employee_id NUMBER) AS
 tot_emps NUMBER;
 BEGIN
 DELETE FROM employees
 WHERE employees.employee_id = remove_emp.employee_id;
 tot_emps := tot_emps - 1;
 END;
/
```

--ALTER PROCEDURE

ALTER PROCEDURE [schema.] procedure_name { procedure_compile_clause | { EDITIONABLE | NONEDITIONABLE } }

example

ALTER PROCEDURE hr.remove_emp COMPILE;

--DROP PROCEDURE

DROP PROCEDURE [schema.] procedure ;

--compile PROCEDURE

ALTER PROCEDURE update_salary COMPILE;

--CREATE FUNCTION

syntax

CREATE [OR REPLACE][EDITIONABLE | NONEDITIONABLE]FUNCTION plsql_function_source

vahidusefzadeh@gmail.com

example

```
CREATE FUNCTION get_bal(acc_no IN NUMBER)
 RETURN NUMBER
 IS acc_bal NUMBER(11,2);
 BEGIN
 SELECT order_total
 INTO acc_bal
 FROM orders
 WHERE customer_id = acc_no;
 RETURN(acc_bal);
 END;
/
```

--ALTER FUNCTION

ALTER FUNCTION [schema.] function_name{ function_compile_clause | { EDITIONABLE | NONEDITIONABLE } }

example

```
ALTER FUNCTION oe.get_bal COMPILE;
```

--DROP FUNCTION

DROP FUNCTION [schema.] function_name ;

--CREATE PACKAGE

syntax

CREATE [OR REPLACE][EDITIONABLE | NONEDITIONABLE] PACKAGE plsql_package_source

example

```
CREATE PACKAGE emp_bonus AS
 PROCEDURE calc_bonus (date_hired employees.hire_date%TYPE);
END emp_bonus;
```

/

--CREATE PACKAGE BODY

syntax

CREATE [OR REPLACE][EDITIONABLE | NONEDITIONABLE]PACKAGE BODY plsql_package_body_source

example

```
CREATE PACKAGE BODY emp_bonus AS
 -- DATE does not match employees.hire_date%TYPE
 PROCEDURE calc_bonus (date_hired DATE) IS
 BEGIN
 DBMS_OUTPUT.PUT_LINE
 ('Employees hired on ' || date_hired || ' get bonus.');
 END;
END emp_bonus;
```

/

--ALTER PACKAGE

ALTER PACKAGE [schema.] package_name { package_compile_clause | { EDITIONABLE | NONEDITIONABLE } }

--DROP PACKAGE

DROP PACKAGE [BODY] [schema.] package ;

--compile PACKAGE

ALTER PACKAGE acct_mgmt COMPILE BODY;

ALTER PACKAGE acct_mgmt COMPILE PACKAGE;

--useful query

```
select l.owner,l.object_name,l.object_type,l.status from dba_objects l where l.status='INVALID' and
l.object_type='PACKAGE';
```

Index

vahidusefzadeh@gmail.com

-- create index

syntax

CREATE [UNIQUE | BITMAP] INDEX [schema.] index ON { cluster_index_clause | table_index_clause | bitmap_join_index_clause }[USABLE | UNUSABLE] ;

example

CREATE INDEX emp_ename ON emp(ename) TABLESPACE users STORAGE (INITIAL 20K NEXT 20k);

CREATE **BITMAP** INDEX gender_idx ON employee(gender) TABLESPACE emp_index_05;

CREATE INDEX reverse_idx ON employee(emp_id) **REVERSE**;

CREATE **UNIQUE** INDEX dept_unique_index ON dept (dname) TABLESPACE indx;

-- Creating an Index Associated with a Constraint

CREATE TABLE emp (empno NUMBER(5) PRIMARY KEY, age INTEGER) ENABLE PRIMARY KEY USING INDEX TABLESPACE users;

-- Creating an Index Online

CREATE INDEX emp_name ON emp (mgr, emp1, emp2, emp3) ONLINE;

-- Creating a Key-Compressed Index

CREATE INDEX usef.emp_ename ON emp(ename) TABLESPACE users COMPRESS 1;

ALTER INDEX usef.emp_ename REBUILD NOCOMPRESS;

-- Creating an Unusable Index

CREATE TABLE employees_part PARTITION BY HASH (employee_id) PARTITIONS 2 AS SELECT * FROM employees;

-- Creating an Invisible Index

CREATE INDEX emp_ename ON emp(ename) TABLESPACE users STORAGE (INITIAL 20K NEXT 20k) INVISIBLE;

-- Creating Multiple Indexes on the Same Set of Columns

CREATE INDEX oe.ord_customer_ix1 ON oe.orders (customer_id, sales_rep_id);

ALTER INDEX oe.ord_customer_ix1 INVISIBLE;

CREATE BITMAP INDEX oe.ord_customer_ix2 ON oe.orders (customer_id, sales_rep_id);

---

vahidusefzadeh@gmail.com

--ALTER INDEX

```
ALTER INDEX [ schema. ]index
 { { deallocate_unused_clause   | allocate_extent_clause   | shrink_clause   | parallel_clause
   | physical_attributes_clause   | logging_clause   | partial_index_clause   } ...
 | rebuild_clause  | PARAMETERS ( 'ODCI_parameters' )  )  | COMPILE  | { ENABLE | DISABLE }
 | UNUSABLE [ ONLINE ]  | VISIBLE | INVISIBLE  | RENAME TO new_name  | COALESCE [ CLEANUP ]
 | { MONITORING | NOMONITORING } USAGE  | UPDATE BLOCK REFERENCES  | alter_index_partitioning  } ;
```

-- Altering Storage Characteristics of an Index

```
ALTER INDEX emp_ename  STORAGE (NEXT 40);

ALTER TABLE emp  ENABLE PRIMARY KEY USING INDEX;
```

-- Rebuilding an Existing Index

```
ALTER INDEX emp_name REBUILD;

ALTER INDEX emp_name REBUILD ONLINE;
```

-- Making an Index Unusable

```
ALTER INDEX emp_email_uk UNUSABLE;

ALTER INDEX i_emp_ename MODIFY PARTITION p2_i_emp_ename UNUSABLE;
```

-- Making an Index Invisible or Visible

```
ALTER INDEX index INVISIBLE;

ALTER INDEX index VISIBLE;
```

--Renaming an Index

```
ALTER INDEX index_name RENAME TO new_name;
```

--Monitoring Index Usage

```
ALTER INDEX index MONITORING USAGE;

ALTER INDEX index NOMONITORING USAGE;
```

-- Monitoring Space Use of Indexes

```
SELECT PCT_USED FROM INDEX_STATS WHERE NAME = 'index';
```

vahidusefzadeh@gmail.com

--Partitioned Indexes

--- Global Indexes

(range):

```
CREATE INDEX ticketsales_idx ON
ticket_sales(month) GLOBAL
PARTITION BY range(month)
(PARTITION ticketsales1_idx VALUES LESS THAN (3)
 PARTITION ticketsales1_idx VALUES LESS THAN (6)
 PARTITION ticketsales2_idx VALUES LESS THAN (9)
 PARTITION ticketsales3_idx VALUES LESS THAN (MAXVALUE));
```

(hash):

```
CREATE INDEX hgidx
ON tab (c1,c2,c3) GLOBAL
PARTITION BY HASH (c1,c2)
(PARTITION p1 TABLESPACE ts1,
 PARTITION p2 TABLESPACE ts2,
 PARTITION p3 TABLESPACE ts3,
 PARTITION p4 TABLESPACE ts4);
```

--- Local Indexes

```
CREATE INDEX ticket_no_idx ON  ticket_sales(ticket__no) LOCAL TABLESPACE localidx_01;
```

-- DROP INDEX

syntax

```
DROP INDEX [ schema. ] index [ ONLINE ] [ FORCE ] ;
```

example

```
DROP INDEX emp_ename;
```

vahidusefzadeh@gmail.com

--info

DBA_INDEXES , DBA_IND_COLUMNS, DBA_IND_EXPRESSIONS, DBA_IND_STATISTICS, INDEX_STATS

VIEW, SEQUENCE, SYNONYM

-- create view

syntax

CREATE [OR REPLACE] [[NO] FORCE] [EDITIONING | EDITIONABLE [EDITIONING] | NONEDITIONABLE]

VIEW [schema.] view [({ alias [VISIBLE | INVISIBLE] [inline_constraint...] | out_of_line_constraint } [, { alias

[VISIBLE | INVISIBLE] [inline_constraint...] | out_of_line_constraint }]) | object_view_clause |

XMLType_view_clause] [BEQUEATH { CURRENT_USER | DEFINER }] AS subquery [subquery_restriction_clause]
;

example

CREATE VIEW sales_staff AS SELECT empno, ename, deptno FROM emp WHERE deptno = 10

WITH CHECK OPTION CONSTRAINT sales_staff_cnst;

--ALTER VIEW

syntax

ALTER VIEW [schema.] view

{ ADD out_of_line_constraint | MODIFY CONSTRAINT constraint { RELY | NORELY }

| DROP { CONSTRAINT constraint | PRIMARY KEY | UNIQUE (column [, column]...) }

| COMPILE | { READ ONLY | READ WRITE } | { EDITIONABLE | NONEDITIONABLE } } ;

example

ALTER VIEW customer_ro COMPILE;

--DROP VIEW

syntax

DROP VIEW [schema.] view [CASCADE CONSTRAINTS] ;

--COMPILE VIEW

ALTER VIEW emp_dept COMPILE;

vahidusefzadeh@gmail.com

--usefule query

```
select view_name from all_views where view_name like '%LOGMNR " ;'
```

--CREATE SEQUENCE

syntax

```
CREATE SEQUENCE [ schema. ] sequence   [ { INCREMENT BY | START WITH } integer
  | { MAXVALUE integer | NOMAXVALUE }   | { MINVALUE integer | NOMINVALUE }
  | { CYCLE | NOCYCLE }   | { CACHE integer | NOCACHE }   | { ORDER | NOORDER }
  | { KEEP | NOKEEP }   | { SESSION | GLOBAL }   ]...;
```

example

```
CREATE SEQUENCE emp_sequence  INCREMENT BY 1  START WITH 1  NOMAXVALUE  NOCYCLE  CACHE 10;
```

--ALTER SEQUENCE

syntex

```
ALTER SEQUENCE [ schema. ] sequence  { INCREMENT BY integer
  | { MAXVALUE integer | NOMAXVALUE }  | { MINVALUE integer | NOMINVALUE }
  | { CYCLE | NOCYCLE }  | { CACHE integer | NOCACHE }  | { ORDER | NOORDER }
  | { KEEP | NOKEEP }  | { SESSION | GLOBAL }  } ...;
```

example

```
ALTER SEQUENCE emp_sequence INCREMENT BY 10 MAXVALUE 10000 CYCLE CACHE 20;
```

--DROP SEQUENCE

syntex

```
DROP SEQUENCE [ schema. ] sequence_name ;
```

-- Creating Synonyms

syntex

```
CREATE [ OR REPLACE ] [ EDITIONABLE | NONEDITIONABLE ]  [ PUBLIC ] SYNONYM  [ schema. ] synonym  FOR [
schema. ] object [ @ dblink ] ;
```

www.usefzadeh.com

example

CREATE PUBLIC SYNONYM public_emp FOR jward.emp;

-- ALTER SYNONYM

syntex

ALTER [PUBLIC] SYNONYM [schema.] synonym { EDITIONABLE | NONEDITIONABLE | COMPILE } ;

example

ALTER SYNONYM offices COMPILE;

ALTER PUBLIC SYNONYM emp_table COMPILE;

ALTER SYNONYM offices NONEDITIONABLE;

--DROP SYNONYM

syntex

DROP [PUBLIC] SYNONYM [schema.] synonym [FORCE] ;

Cluster

-- create cluster

syntax

CREATE CLUSTER [schema.] cluster (column datatype [SORT] [, column datatype [SORT]]...)

[{ physical_attributes_clause | SIZE size_clause | TABLESPACE tablespace | { INDEX

| [SINGLE TABLE] HASHKEYS integer [HASH IS expr] } }...] [parallel_clause]

[NOROWDEPENDENCIES | ROWDEPENDENCIES] [CACHE | NOCACHE] ;

example

CREATE CLUSTER emp_dept (deptno NUMBER(3)) SIZE 600 TABLESPACE users

STORAGE (INITIAL 200K NEXT 300K MINEXTENTS 2 MAXEXTENTS 20 PCTINCREASE 33);

CREATE TABLE emp (empno NUMBER(5) PRIMARY KEY, ename VARCHAR2(15) NOT NULL, deptno NUMBER(3) REFERENCES dept) **CLUSTER** emp_dept (deptno);

CREATE TABLE dept (deptno NUMBER(3) PRIMARY KEY, . . .) **CLUSTER** emp_dept (deptno);

vahidusefzadeh@gmail.com

-- create cluster index

```
CREATE INDEX emp_dept_index  ON CLUSTER emp_dept  TABLESPACE users

 STORAGE (INITIAL 50K  NEXT 50K  MINEXTENTS 2  MAXEXTENTS 10  PCTINCREASE 33);
```

-- Creating Hash Clusters

```
CREATE CLUSTER trial_cluster (trialno NUMBER(5,0))  TABLESPACE users

 STORAGE (INITIAL 250K NEXT 50K  MINEXTENTS 1 MAXEXTENTS 3  PCTINCREASE 0)  HASH IS trialno HASHKEYS
150;

CREATE TABLE trial ( trialno NUMBER(5,0) PRIMARY KEY) CLUSTER trial_cluster (trialno);
```

-- Creating a Sorted Hash Cluster

```
CREATE CLUSTER call_detail_cluster ( telephone_number NUMBER, call_timestamp NUMBER SORT, call_duration
NUMBER SORT )

 HASHKEYS 10000 HASH IS telephone_number  SIZE 256;

CREATE TABLE call_detail (  telephone_number NUMBER,  call_timestamp NUMBER SORT,  call_duration NUMBER
SORT,  other_info VARCHAR2(30) ) CLUSTER call_detail_cluster ( telephone_number, call_timestamp, call_duration
);
```

-- Creating Single-Table Hash Clusters

```
CREATE CLUSTER peanut (variety NUMBER)

SIZE 512 SINGLE TABLE HASHKEYS 500;
```

-- DROP CLUSTER

syntax

```
DROP CLUSTER [ schema. ] cluster   [ INCLUDING TABLES [ CASCADE CONSTRAINTS ] ] ;
```

example

```
DROP CLUSTER emp_dept;

DROP CLUSTER emp_dept INCLUDING TABLES;

DROP CLUSTER emp_dept INCLUDING TABLES CASCADE CONSTRAINTS;
```

---

vahidusefzadeh@gmail.com

DBMS_SCHEDULER

--Creating a Job

```
BEGIN
 DBMS_SCHEDULER.CREATE_JOB (
  job_name         =>  'update_sales',
  job_type         =>  'STORED_PROCEDURE',
  job_action       =>  'OPS.SALES_PKG.UPDATE_SALES_SUMMARY',
  start_date       =>  '28-APR-08 07.00.00 PM Australia/Sydney',
  repeat_interval    =>  'FREQ=DAILY;INTERVAL=2', /* every other day */
  end_date         =>  '20-NOV-08 07.00.00 PM Australia/Sydney',
  auto_drop        =>   FALSE,
  job_class        =>  'batch_update_jobs',
  comments         =>  'My new job');
END;
/
```

-- Setting Repeat Intervals

FREQ ➔ YEARLY, MONTHLY, WEEKLY, DAILY, HOURLY, MINUTELY, and SECONDLY.

BYMONTH 1-12 1,4,6

BYYEARDAY any positive or negative number

BYMONTHDAY any positive or negative number (eg -1 last day of the month)

BYDAY (MON, TUE, and so on) can be prefixed with a number -1FRI

BYHOUR 0-23 , BYMINUTE 0-59 , BYSECOND 0-59

examples:

Every Monday ➔ FREQ=WEEKLY; BYDAY=MON;

Every other ➔ Monday FREQ=WEEKLY; BYDAY=MON; INTERVAL=2;

Last day of each month ➔ FREQ=MONTHLY; BYMONTHDAY=-1;

Every January 7 ➔ FREQ=YEARLY; BYMONTH=JAN; BYMONTHDAY=7;

Second Wednesday of ➔ FREQ=MONTHLY; BYDAY=2WED;

each month

Every hour ➔ FREQ=HOURLY;

Every 4 hours ➔ FREQ=HOURLY; INTERVAL=4;

Hourly on the first ➔ FREQ=HOURLY; BYMONTHDAY=1;

day of each month

15th day of every other ➔ FREQ=MONTHLY; BYMONTHDAY=15; INTERVAL=2

--Creating Jobs Using a Named Program

```
BEGIN
 DBMS_SCHEDULER.CREATE_JOB (
  job_name        =>  'my_new_job1',
  program_name     =>  'my_saved_program',
  repeat_interval   =>  'FREQ=DAILY;BYHOUR=12',
  comments        =>  'Daily at noon');
END;
/
```

--Creating Jobs Using a Named Program and Job Styles

```
BEGIN
 DBMS_SCHEDULER.CREATE_JOB (
  job_name       =>  'my_lightweight_job1',
  program_name    =>  'polling_prog_n2',
  repeat_interval  =>  'FREQ=SECONDLY;INTERVAL=10',
  end_date       =>  '30-APR-09 04.00.00 AM Australia/Sydney',
  job_style      => 'LIGHTWEIGHT',
  comments       => 'Job that polls device n2 every 10 seconds');
END;
```

```
/
```

--Creating Jobs Using a Named Schedule

```
BEGIN
 DBMS_SCHEDULER.CREATE_JOB (
  job_name             => 'my_new_job2',
  job_type             => 'PLSQL_BLOCK',
  job_action           => 'BEGIN SALES_PKG.UPDATE_SALES_SUMMARY; END;',
  schedule_name        => 'my_saved_schedule');
END;
/
```

--Creating Jobs Using Named Programs and Schedules

```
BEGIN
 DBMS_SCHEDULER.CREATE_JOB (
  job_name         => 'my_new_job3',
  program_name     => 'my_saved_program1',
  schedule_name    => 'my_saved_schedule1');
END;
/
```

--Creating a Credential

```
BEGIN
 DBMS_CREDENTIAL.CREATE_CREDENTIAL('DW_CREDENTIAL', 'dwuser', 'dW001515');
END;
/
GRANT EXECUTE ON DW_CREDENTIAL TO salesuser;
```

--Creating a Database Destination

```
BEGIN
```

```
 DBMS_SCHEDULER.CREATE_DATABASE_DESTINATION (
  destination_name     => 'DBHOST1_USEFDBDW',
  agent              => 'DBHOST1',
  tns_name           => 'USEFDBDW',
  comments           => 'Instance named usefdbdw on host dbhost1.example.com');
END;
/
```

--Creating a Database Destination Group

```
BEGIN
  DBMS_SCHEDULER.CREATE_GROUP(
    GROUP_NAME    => 'all_dbs',
    GROUP_TYPE    => 'DB_DEST',
    MEMBER        => 'oltp_admin@usefdb, usefdbdw1, LOCAL',
    COMMENTS      => 'All databases managed by me');
END;
/
```

-- Creating a Remote Database Job

```
BEGIN
 DBMS_SCHEDULER.CREATE_JOB (
   job_name           =>  'SALES_SUMMARY1',
   job_type           =>  'STORED_PROCEDURE',
   job_action         =>  'SALES.SALES_REPORT1',
   start_date         =>  '15-JUL-09 11.00.00 PM Europe/Warsaw',
   repeat_interval    =>  'FREQ=DAILY',
   credential_name    =>  'DW_CREDENTIAL',
   destination_name   =>  'DBHOST1_USEFDBDW');
END;
```

/

--Setting Job Arguments

```
BEGIN
 DBMS_SCHEDULER.SET_JOB_ARGUMENT_VALUE (
  job_name             => 'ops_reports',
  argument_position    => 2,
  argument_value       => '12-DEC-03');
END;
/
```

--Altering Jobs

```
BEGIN
 DBMS_SCHEDULER.SET_ATTRIBUTE (
  name       =>  'update_sales',
  attribute  =>  'repeat_interval',
  value      =>  'freq=weekly; byday=wed');
END;
```

--Running Jobs

```
BEGIN
 DBMS_SCHEDULER.RUN_JOB(
   JOB_NAME          => 'DSS.ETLJOB1, DSS.ETLJOB2',
   USE_CURRENT_SESSION => FALSE);
END;
/
```

-- Copying Jobs

```
begin
DBMS_SCHEDULER.COPY_JOB (
```

```
 old_job =>'MY_OLD_JOB',
 new_job =>'MY_NEW_JOB');
end;
/
```

--Stopping Jobs

```
BEGIN
  DBMS_SCHEDULER.STOP_JOB('job1, sys.dw_jobs, 984, 1223');
END;
/
```

--Dropping Jobs

```
BEGIN
  DBMS_SCHEDULER.DROP_JOB ('job1, job3, sys.jobclass1, sys.jobclass2');
END;
/
```

--Disabling Jobs

```
BEGIN
  DBMS_SCHEDULER.DISABLE('job1, job2, job3, sys.jobclass1, sys.jobclass2');
END;
Enabling Jobs
BEGIN
 DBMS_SCHEDULER.ENABLE ('job1, job2, job3, sys.jobclass1, sys.jobclass2, sys.jobclass3');
END;
/
```

--Creating Programs

```
BEGIN
  DBMS_SCHEDULER.CREATE_PROGRAM (
```

```
  program_name         => 'my_program1',
  program_action       => '/usr/local/bin/date',
  program_type         => 'EXECUTABLE',
  comments             => 'My comments here');
END;
/
```

--Defining Program Arguments

```
BEGIN
 DBMS_SCHEDULER.DEFINE_PROGRAM_ARGUMENT (
  program_name          => 'operations_reporting',
  argument_position     => 2,
  argument_name         => 'end_date',
  argument_type         => 'VARCHAR2',
  default_value         => '12-DEC-03');
END;
/
```

--Altering Programs

```
BEGIN
 DBMS_SCHEDULER.SET_ATTRIBUTE (
  name         => 'my_program1',
  attribute    => 'program_action',
  value        => '/usr/local/bin/salesreports1');
END;
/
```

--Dropping Programs

```
BEGIN
 DBMS_SCHEDULER.DROP_PROGRAM('program1, program2, program3');
```

```
END;
/
```

--Enabling Programs

```
BEGIN
  DBMS_SCHEDULER.ENABLE('program1, program2, program3');
END;
/
```

-- create a window

```
BEGIN
   DBMS_SCHEDULER.CREATE_WINDOW (
     window_name      => 'daytime',
     resource_plan    => 'mixed_workload_plan',
     start_date       => '28-APR-09 08.00.00 AM',
     repeat_interval  => 'freq=daily; byday=mon,tue,wed,thu,fri',
     duration         => interval '9' hour,
     window_priority  => 'low',
     comments         => 'OLTP transactions have priority');
END;
/
```

-- Altering Windows

```
BEGIN
dbms_scheduler.set_attribute(
 name => 'MYWINDOW',
 attribute => 'window_priority',
 value => 'LOW');
END;
```

```
/
```

-- closing a window

```
begin
 dbms_scheduler.close_window ('WORK_HOURS_WINDOW');
end;
/
```

--Dropping Windows

```
BEGIN
  DBMS_SCHEDULER.DROP_WINDOW ('window1, window2, window3,
   windowgroup1, windowgroup2');
END;
/
```

--Disabling Windows

```
BEGIN
  DBMS_SCHEDULER.DISABLE ('sys.window1, sys.window2,
   sys.window3, sys.windowgroup1, sys.windowgroup2');
END;
/
```

--Enabling Windows

```
BEGIN
  DBMS_SCHEDULER.ENABLE ('sys.window1, sys.window2, sys.window3');
END;
/
```

--Creating Window Groups

```
BEGIN
  DBMS_SCHEDULER.CREATE_GROUP (
```

```
  group_name   =>  'downtime',
  group_type   =>  'WINDOW',
  member       =>  'weeknights, weekends');
END;
/
```

--Dropping Window Groups

```
BEGIN
DBMS_SCHEDULER.DROP_GROUP('sys.windowgroup1, sys.windowgroup2, sys.windowgroup3');
END;
/
```

--Adding a Member to a Window Group

```
BEGIN
 DBMS_SCHEDULER.ADD_GROUP_MEMBER ('sys.windowgroup1','window2, window3');
END;
/
```

--Removing a Member from a Window Group

```
BEGIN
 DBMS_SCHEDULER.REMOVE_GROUP_MEMBER('sys.window_group1', 'window2, window3');
END;
/
```

--Enabling a Window Group

```
BEGIN
 DBMS_SCHEDULER.ENABLE('sys.windowgroup1, sys.windowgroup2, sys.windowgroup3');
END;
/
```

--Disabling a Window Group

```
BEGIN
```

```
 DBMS_SCHEDULER.DISABLE('sys.windowgroup1, sys.windowgroup2, sys.windowgroup3');
END;
/
```

-- Creating Schedules

```
BEGIN
DBMS_SCHEDULER.CREATE_SCHEDULE (
 schedule_name => 'my_stats_schedule',
 start_date => SYSTIMESTAMP,
 end_date => SYSTIMESTAMP + INTERVAL '30' day,
 repeat_interval => 'FREQ=HOURLY; INTERVAL=4',
 comments => 'Every 4 hours');
END;
/
```

-- altering schedules

```
BEGIN
dbms_scheduler.set_attribute(
 name => 'my_stats_schedule',
 attribute => 'REPEAT_INTERVAL',
 value => 'FREQ=HOURLY; INTERVAL=2');
END;
/
```

-- Dropping Schedules

```
BEGIN
DBMS_SCHEDULER.DROP_SCHEDULE (
 schedule_name  => 'my_stats_schedule',
 force => FALSE); -- DEFAULT
```

vahidusefzadeh@gmail.com

```
END;
/
```

-- Creating Job Classes

```
BEGIN
 DBMS_SCHEDULER.CREATE_JOB_CLASS (
  job_class_name => 'finance_jobs_class',
  resource_consumer_group => 'finance_group');
END;
/
 BEGIN
dbms_scheduler.set_attribute(
 name => ' finance_jobs_class',
 attribute => 'logging_level',
 value => DBMS_SCHEDULER.LOGGING_FAILED_RUNS);
END;
/
```

-- Dropping Job Classes

```
BEGIN
 DBMS_SCHEDULER.DROP_JOB_CLASS('jobclass1, jobclass2, jobclass3');
END;
/
```

--useful query

---Viewing the Currently Active Window and Resource Plan

```
SELECT WINDOW_NAME, RESOURCE_PLAN  FROM DBA_SCHEDULER_WINDOWS WHERE ACTIVE='TRUE';
```

---Finding Information About Currently Running Jobs

```
SELECT JOB_NAME, STATE FROM DBA_SCHEDULER_JOBS WHERE JOB_NAME = 'MY_EMP_JOB1';
```

```
SELECT JOB_NAME , STATE, START_DATE, END_DATE, LAST_START_DATE, NEXT_RUN_DATE,
LAST_RUN_DURATION, FAILURE_COUNT FROM USER_SCHEDULER_JOBS ORDER BY  START_DATE DESC;
```

---progress of currently running jobs

```
SELECT * FROM ALL_SCHEDULER_RUNNING_JOBS;
```

---job that is part of a running chain

```
SELECT * FROM ALL_SCHEDULER_RUNNING_CHAINS  WHERE JOB_NAME='MY_JOB1';
```

---Monitoring and Managing Window and Job Logs

```
SELECT JOB_NAME, OPERATION, OWNER FROM DBA_SCHEDULER_JOB_LOG;
```

---Job Run History Details

```
select log_id,  job_name, status, to_char(log_date, 'DD-MON-YYYY HH24:MI') log_date  from
dba_scheduler_job_run_details where job_name = 'MY_JOB14';
```

---Window Logs

```
SELECT LOG_ID, TO_CHAR(LOG_DATE, 'MM/DD/YYYY'), WINDOW_NAME, OPERATION FROM
DBA_SCHEDULER_WINDOW_LOG;
SELECT LOG_ID, WINDOW_NAME, ACTUAL_START_DATE, ACTUAL_DURATION FROM
DBA_SCHEDULER_WINDOW_DETAILS;
```

PDB

--CREATE PLUGGABLE DATABASE

syntax

```
CREATE PLUGGABLE DATABASE pdb_name  { create_pdb_from_seed | create_pdb_clone | create_pdb_from_xml }
;
```

examples

```
CREATE PLUGGABLE DATABASE salespdb ADMIN USER salesadm IDENTIFIED BY password;
```

OR

```
create pluggable database pdb_clone from PDB1
```

file_name_convert=('/u01/oracle/oradata/USEF_CDB/pdb1/datafile','/u01/oracle/oradata/USEF_CDB/pdb

_clone/datafile');

--Connecting to a PDB:

sqlplus "sys/sys@192.168.232.10:1521/pdb1 as sysdba"

connect sys/sys@192.168.232.10:1521/pdb1 as sysdba;

TNS:

PDB1 =

(DESCRIPTION =(ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.232.10)(PORT = 1521))

(CONNECT_DATA =

(SERVER = DEDICATED)

(service_name = pdb1)))

sqlplus "sys/sys@pdb1 as sysdba"

--Creating Local User in a PDB

CREATE USER testpdb IDENTIFIED BY password DEFAULT TABLESPACE pdb1_tbs QUOTA UNLIMITED ON pdb1_tbs CONTAINER = CURRENT;

--Creating global User

create user c##usef identified by jjjj;

--ALTER PLUGGABLE DATABASE

ALTER PLUGGABLE DATABASE { pdb_unplug_clause | pdb_settings_clauses

| pdb_datafile_clause | pdb_recovery_clauses | pdb_change_state | pdb_change_state_from_root } ;

---Changing Open Mode

ALTER PLUGGABLE DATABASE <pdb_name> OPEN READ [WRITE] [UPGRADE] [RESTRICTED] [FORCE] [INSTANCES = <('<instance_name>' | <ALL [EXCEPT ('<instance_name'>)] >);

ALTER PLUGGABLE DATABASE salespdb OPEN READ ONLY RESTRICTED;

ALTER PLUGGABLE DATABASE ALL OPEN READ WRITE FORCE;

ALTER PLUGGABLE DATABASE ALL OPEN READ WRITE;

ALTER PLUGGABLE DATABASE OPEN READ ONLY;

ALTER PLUGGABLE DATABASE OPEN FORCE;

ALTER PLUGGABLE DATABASE ALL EXCEPT salespdb, hrpdb CLOSE IMMEDIATE;

ALTER PLUGGABLE DATABASE CLOSE IMMEDIATE;

ALTER PLUGGABLE DATABASE pdb5 OPEN READ WRITE INSTANCES = ('ORCLDB_1', 'ORCLDB_2');

ALTER PLUGGABLE DATABASE pdb6 CLOSE RELOCATE TO 'ORCLDB_3';

--- Changing Default Tablespaces

ALTER PLUGGABLE DATABASE DATAFILE '/u03/oracle/pdb1_01.dbf' ONLINE;

--- Setting Storage Limit

ALTER PLUGGABLE DATABASE <pdb_name> STORAGE (MAX_SHARED_TEMP_SIZE <UNLIMITED | <size_clause>);

ALTER PLUGGABLE DATABASE DEFAULT TABLESPACE pdb1_tbs;

ALTER PLUGGABLE DATABASE DEFAULT TEMPORARY TABLESPACE pdb1_temp;

ALTER PLUGGABLE DATABASE STORAGE(MAXSIZE 2G);

ALTER PLUGGABLE DATABASE STORAGE(MAXSIZE UNLIMITED);

ALTER PLUGGABLE DATABASE STORAGE UNLIMITED;

---Setting Default Edition

ALTER PLUGGABLE DATABASE DEFAULT EDITION = PDB1E3;

--Database Files Create Datafile

ALTER PLUGGABLE DATABASE <pdb_name> CREATE DATAFILE <'<datafile_name>' | filenumber> [AS <file_specification> | NEW]

--Database Files Alter Datafile

ALTER PLUGGABLE DATABASE <pdb_name> DATAFILE <'<datafile_name>' | filenumber> <ONLINE | OFFLINE [FOR DROP] | RESIZE <size_clause> | <autoextend_clause> | END BACKUP>;

--Database Files Alter Tempfile

ALTER PLUGGABLE DATABASE <pdb_name> TEMPFILE <'<datafile_name>' | filenumber> <RESIZE | <autoextend_clause> | DROP [INCLUDING DATAFILES] | ONLINE | OFFLINE>;

--Database Files Move Datafile

ALTER PLUGGABLE DATABASE <pdb_name> <move_datafile_clause> MOVE DATAFILE <'<file_name>' | ASM_file_name | filenumber> TO '<file_name | ASM_file_name>' [REUSE] [KEEP]

--Supplemental Logging

ALTER PLUGGABLE DATABASE <pdb_name> <ADD | DROP> SUPPLEMENTAL LOG <DATA | <supplemental_id_key_clause> | <supplemental_plsql_clause>>

ALTER PLUGGABLE DATABASE pdborcl ADD SUPPLEMENTAL LOG DATA;

---change global database name

ALTER PLUGGABLE DATABASE RENAME GLOBAL_NAME TO salespdb.example.com;

--Unplugging PDB

ALTER PLUGGABLE DATABASE salespdb UNPLUG INTO '/oracle/data/salespdb.xml';

--RMAN

backup

rman target “sys/sys@pdb1”

RMAN> backup database format ‘/u01/oracle/usef_backup/%U.bkp’;

restore

rman target /

run{

restore pluggable database PDB1;

recover pluggable database PDB1;

}

-- ALTER SYSTEM on a PDB

ALTER SYSTEM FLUSH SHARED_POOL;

ALTER SYSTEM FLUSH BUFFER_CACHE;

ALTER SYSTEM ENABLE RESTRICTED SESSION;

ALTER SYSTEM DISABLE RESTRICTED SESSION;

ALTER SYSTEM SET USE_STORED_OUTLINES;

ALTER SYSTEM SUSPEND;

ALTER SYSTEM RESUME;

ALTER SYSTEM CHECKPOINT;

ALTER SYSTEM CHECK DATAFILES;

ALTER SYSTEM REGISTER;

ALTER SYSTEM KILL SESSION;

ALTER SYSTEM DISCONNECT SESSION;

ALTER SYSTEM SET initialization_parameter (for a subset of initialization parameters);

alter session set container=PDB1;

--DROP PLUGGABLE DATABASE

DROP PLUGGABLE DATABASE pdb_name [{ KEEP | INCLUDING } DATAFILES] ;

example

alter pluggable database CONT1PLUG1 close;

drop pluggable database CONT1PLUG1 including datafiles;

Materialized View

--CREATE MATERIALIZED VIEW

Syntax

```
CREATE MATERIALIZED VIEW [ schema. ] materialized_view [ column_alias [ENCRYPT [encryption_spec]] [,
column_alias [ENCRYPT [encryption_spec]] ]... ] [ OF [ schema. ] object_type ] [ (scoped_table_ref_constraint) ]
 { ON PREBUILT TABLE [ { WITH | WITHOUT } REDUCED PRECISION ] | physical_properties
materialized_view_props } [ USING INDEX [ physical_attributes_clause | TABLESPACE tablespace
 ]... | USING NO INDEX ] [ create_mv_refresh ] [ FOR UPDATE ] [ evaluation_edition_clause ]
 [ query_rewrite_clause ] AS subquery ;
```

```
CREATE MATERIALIZED VIEW view-name
```

vahidusefzadeh@gmail.com

```
BUILD [IMMEDIATE | DEFERRED]

REFRESH [FAST | COMPLETE | FORCE ]

ON [COMMIT | DEMAND ]

[[ENABLE | DISABLE] QUERY REWRITE]

[ON PREBUILT TABLE] AS   SELECT ...;
```

example

--Fast Refresh

Syntax

```
CREATE MATERIALIZED VIEW <schema.name> PCTFREE <integer> PCTUSED <integer> TABLESPACE

<tablespace_name> BUILD IMMEDIATE REFRESH <FAST | FORCE> ON <COMMIT | DEMAND> <USING INDEX |

USING NO INDEX> INITRANS <integer> STORAGE CLAUSE AS (<SQL statement>);
```

Example:

```
CREATE MATERIALIZED VIEW mv_simple TABLESPACE uwdata BUILD IMMEDIATE REFRESH FAST ON COMMIT AS
SELECT * FROM servers;
```

--Force Refresh

Syntax

```
CREATE MATERIALIZED VIEW <schema.name> PCTFREE <integer> PCTUSED <integer> TABLESPACE

<tablespace_name> BUILD IMMEDIATE REFRESH <FAST | FORCE> ON <COMMIT | DEMAND>  AS (<SQL

statement>);
```

Example:

```
CREATE MATERIALIZED VIEW mv_force
TABLESPACE uwdata
NOCACHE
LOGGING
NOCOMPRESS
NOPARALLEL
BUILD IMMEDIATE
REFRESH FORCE ON DEMAND
WITH ROWID AS
SELECT * FROM servers;
```

vahidusefzadeh@gmail.com

--Complete Refresh

Syntax

```
CREATE MATERIALIZED VIEW <schema.name> PCTFREE <integer> PCTUSED <integer>

TABLESPACE <tablespace_name> REFRESH <COMPLETE | FORCE> START WITH <date>

NEXT <date_calculation> [FOR UPDATE] AS (<SQL statement>);
```

Example:

```
CREATE MATERIALIZED VIEW mv_complete
TABLESPACE uwdata
REFRESH COMPLETE
START WITH SYSDATE
NEXT SYSDATE + 1
AS SELECT s.srvr_id, i.installstatus, COUNT(*)
FROM servers s, serv_inst i
WHERE s.srvr_id = i.srvr_id
GROUP BY s.srvr_id, i.installstatus;
```

--Complete Refresh Using Index

Syntax

```
CREATE MATERIALIZED VIEW <schema.name> [LOGGING] [CACHE] PCTFREE <integer> PCTUSED <integer>

USING INDEX TABLESPACE <tablespace_name> REFRESH <COMPLETE | FORCE> START WITH <date>

NEXT <date_calculation> [FOR UPDATE] AS (<SQL statement>);
```

Example:

```
CREATE SNAPSHOT mv_w_index
LOGGING CACHE
PCTFREE 0 PCTUSED 99
TABLESPACE uwdata
REFRESH COMPLETE
AS SELECT s.srvr_id, COUNT(*)
FROM servers s, serv_inst i
WHERE s.srvr_id = i.srvr_id
GROUP BY s.srvr_id;
```

--Prebuilt Table

Syntax

```
CREATE MATERIALIZED VIEW <schema.name> PCTFREE <integer> PCTUSED <integer> TABLESPACE
```

www.usefzadeh.com

```
<tablespace_name> REFRESH <COMPLETE | FORCE> START WITH <date> NEXT <date_calculation>

[FOR UPDATE] AS (<SQL statement>);
```

Example:

```
CREATE TABLE mv_prebuilt (
month VARCHAR2(8),
state VARCHAR2(40),
sales NUMBER(10,2));

CREATE MATERIALIZED VIEW mv_prebuilt
ON PREBUILT TABLE WITH REDUCED PRECISION
AS SELECT t.calendar_month_desc AS month,
c.cust_state_province AS state,
SUM(s.amount_sold) AS sales
FROM times t, customers c, sales s
WHERE s.time_id = t.time_id AND s.cust_id = c.cust_id
GROUP BY t.calendar_month_desc, c.cust_state_province;
```

--Enable Query Rewrite

Syntax

```
CREATE MATERIALIZED VIEW <schema.name> PCTFREE <integer> PCTUSED <integer> TABLESPACE

<tablespace_name> REFRESH <COMPLETE | FORCE> START WITH <date> NEXT <date_calculation>

[FOR UPDATE] AS (<SQL statement>);
```

Example:

```
SELECT name, value FROM gv$parameter WHERE name LIKE '%rewrite%';

EXPLAIN PLAN FOR SELECT s.srvr_id, i.installstatus, COUNT(*) FROM servers s, serv_inst i
WHERE s.srvr_id = i.srvr_id AND s.srvr_id = 502 GROUP BY s.srvr_id, i.installstatus;

SELECT * FROM TABLE(dbms_xplan.display);

CREATE MATERIALIZED VIEW mv_rewrite TABLESPACE uwdata REFRESH ON DEMAND ENABLE QUERY REWRITE
AS SELECT s.srvr_id, i.installstatus, COUNT(*) FROM servers s, serv_inst I WHERE s.srvr_id = i.srvr_id GROUP
BY s.srvr_id, i.installstatus;

EXPLAIN PLAN FOR SELECT s.srvr_id, i.installstatus, COUNT(*) FROM servers s, serv_inst i
WHERE s.srvr_id = i.srvr_id AND s.srvr_id = 502 GROUP BY s.srvr_id, i.installstatus;


SELECT * FROM TABLE(dbms_xplan.display);
```

---

vahidusefzadeh@gmail.com

-- CREATE MATERIALIZED VIEW LOG

syntax

```
CREATE MATERIALIZED VIEW LOG ON [ schema. ] table  [ physical_attributes_clause

| TABLESPACE tablespace  | logging_clause  | { CACHE | NOCACHE }  ]...  [ parallel_clause ]  [

table_partitioning_clauses ]  [ WITH [ { OBJECT ID  | PRIMARY KEY  | ROWID | SEQUENCE  | COMMIT SCN  }  [ {

, OBJECT ID  | , PRIMARY KEY  | , ROWID  | , SEQUENCE | , COMMIT SCN  }  ]... ]  (column [, column ]...)  [

new_values_clause ] ] [ mv_log_purge_clause ] [ for_refresh_clause ];
```

example

```
CREATE MATERIALIZED VIEW LOG ON sales WITH ROWID;

CREATE MATERIALIZED VIEW LOG ON sales WITH ROWID (prod_id, cust_id, time_id, channel_id, promo_id,

quantity_sold, amount_sold) INCLUDING NEW VALUES;
```

--ALTER MATERIALIZED VIEW LOG

```
ALTER MATERIALIZED VIEW LOG [ FORCE ]  ON [ schema. ]table  [ physical_attributes_clause

| add_mv_log_column_clause  | alter_table_partitioning  | parallel_clause  | logging_clause

| allocate_extent_clause  | shrink_clause  | move_mv_log_clause  | { CACHE | NOCACHE }  ]

 [ mv_log_augmentation ] [  mv_log_purge_clause ] [ for_refresh_clause ]  ;
```

--Materialized View Log Purging

```
ALTER MATERIALIZED VIEW LOG ON sales PURGE IMMEDIATE;

CREATE MATERIALIZED VIEW LOG ON sales PURGE START WITH sysdate NEXT sysdate+1 WITH ROWID (prod_id,

cust_id, time_id, channel_id, promo_id, quantity_sold, amount_sold) INCLUDING NEW VALUES;
```

-- Shrink Log

```
ALTER MATERIALIZED VIEW LOG [FORCE] ON <schema.table_name> [COMPACT] [CASCADE];

ALTER MATERIALIZED VIEW LOG ON servers SHRINK SPACE COMPACT CASCADE;

ALTER TABLE mlog$_servers ENABLE ROW MOVEMENT;

ALTER MATERIALIZED VIEW LOG ON servers SHRINK SPACE COMPACT CASCADE;
```

vahidusefzadeh@gmail.com

-- Alter Parallel Access

ALTER MATERIALIZED VIEW LOG [FORCE] ON <schema.table_name> <NOPARALLEL | PARALLEL <integer>>;

ALTER MATERIALIZED VIEW LOG ON servers PARALLEL 8;

-- Alter Logging Clause

ALTER MATERIALIZED VIEW LOG [FORCE] ON <schema.table_name> <LOGGING | NOLOGGING>;

ALTER MATERIALIZED VIEW LOG ON servers LOGGING;

-- Alter Allocate Extent by Size

ALTER MATERIALIZED VIEW LOG [FORCE] ON <schema.table_name> ALLOCATE EXTENT (SIZE <integer> <M | G | T>);

ALTER MATERIALIZED VIEW LOG ON servers ALLOCATE EXTENT (SIZE 512K);

--Alter Allocate Extent by Datafile

ALTER MATERIALIZED VIEW LOG [FORCE] ON <schema.table_name> ALLOCATE EXTENT (DATAFILE <file_name>);

ALTER MATERIALIZED VIEW LOG ON servers ALLOCATE EXTENT (DATAFILE 'u01/oracle/usefdb/system01.dbf');

-- Alter Allocate Extent by Instance

ALTER MATERIALIZED VIEW LOG [FORCE] ON <schema.table_name> ALLOCATE EXTENT SIZE (INSTANCE <integer>);

ALTER MATERIALIZED VIEW LOG ON servers ALLOCATE EXTENT (INSTANCE 1);

--Log Caching

ALTER MATERIALIZED VIEW LOG [FORCE] ON <schema.table_name> <CACHE | NOCACHE>;

ALTER MATERIALIZED VIEW LOG ON servers CACHE;

--DROP MATERIALIZED VIEW LOG

syntax

DROP MATERIALIZED VIEW LOG ON [schema.] table ;

--Choosing Indexes for Materialized Views

CREATE UNIQUE INDEX ."I_SNAP$_" ON . (SYS_OP_MAP_NONNULL("LOG_DATE")) PCTFREE 10 INITRANS 2

MAXTRANS 255 COMPUTE STATISTICS STORAGE(INITIAL 65536 NEXT 1048576 MINEXTENTS 1 MAXEXTENTS

2147483645 PCTINCREASE 0 FREELISTS 1 FREELIST GROUPS 1 BUFFER_POOL DEFAULT FLASH_CACHE DE FAULT

CELL_FLASH_CACHE DEFAULT) TABLESPACE ;

--ALTER MATERIALIZED VIEW

syntax

ALTER MATERIALIZED VIEW [schema.] materialized_view [physical_attributes_clause

| modify_mv_column_clause | table_compression | LOB_storage_clause [, LOB_storage_clause]...

| modify_LOB_storage_clause [, modify_LOB_storage_clause]... | alter_table_partitioning

| parallel_clause | logging_clause | allocate_extent_clause | deallocate_unused_clause

| shrink_clause | { CACHE | NOCACHE }] [alter_iot_clauses] [USING INDEX physical_attributes_clause]

[MODIFY scoped_table_ref_constraint | alter_mv_refresh] [evaluation_edition_clause]

[alter_query_rewrite_clause | COMPILE | CONSIDER FRESH] ;

Examples

ALTER MATERIALIZED VIEW sales_by_month_by_state REFRESH FAST;

ALTER MATERIALIZED VIEW sales_by_month_by_state REFRESH NEXT SYSDATE+7;

ALTER MATERIALIZED VIEW sales_by_month_by_state CONSIDER FRESH;

ALTER MATERIALIZED VIEW emp_data REFRESH COMPLETE START WITH TRUNC(SYSDATE+1) + 9/24 NEXT SYSDATE+7;

--Allocate Extent

ALTER MATERIALIZED VIEW <schema.materialized_view> ALLOCATE EXTENT (SIZE <size_clause> DATAFILE <'file_name'>)

INSTANCE <integer>;

ALTER MATERIALIZED VIEW mv_simple ALLOCATE EXTENT (SIZE 64K);

--Caching

ALTER MATERIALIZED VIEW <schema.materialized_view> <CACHE | NOCACHE>;

```
ALTER MATERIALIZED VIEW mv_simple CACHE;

--Coalesce

ALTER MATERIALIZED VIEW <schema.materialized_view> <index_organized_table_clause>
<alter_overflow_clause> <alter_mapping_table_clause> COALESCE;

--Compile

ALTER MATERIALIZED VIEW <schema.materialized_view> COMPILE;

ALTER MATERIALIZED VIEW mv_simple COMPILE;

--Consider Fresh

ALTER MATERIALIZED VIEW <schema.materialized_view> CONSIDER FRESH;

ALTER MATERIALIZED VIEW mv_complete CONSIDER FRESH;

--Deallocate Unused

ALTER MATERIALIZED VIEW <schema.materialized_view> DEALLOCATE UNUSED;

ALTER MATERIALIZED VIEW mv_complete DEALLOCATE UNUSED;

--LOB Storage

ALTER MATERIALIZED VIEW <schema.materialized_view> LOB (lob_item) STORE AS (lob_storage_parameters);

--Logging

ALTER MATERIALIZED VIEW <schema.materialized_view> <LOGGING | NOLOGGING>;

ALTER MATERIALIZED VIEW mv_simple LOGGING;

--Modify LOB Storage

ALTER MATERIALIZED VIEW <schema.materialized_view> MODIFY LOB (<lob_item>) (new_lob_parameter);

--Parallel Access

ALTER MATERIALIZED VIEW <schema.materialized_view> <PARALLEL | NO_PARALLEL>;

ALTER MATERIALIZED VIEW mv_simple PARALLEL;

--Physical Attributes
```

ALTER MATERIALIZED VIEW <schema.materialized_view> (PCT_FREE <integer> PCT_USED <integer> INITRANS <integer> TABLESPACE <tablespace_name>;

ALTER MATERIALIZED VIEW mv_simple PCTFREE 1;

--Query Rewrite

ALTER MATERIALIZED VIEW <schema.materialized_view> <ENABLE | DISABLE> QUERY REWRITE;

ALTER MATERIALIZED VIEW mv_simple ENABLE QUERY REWRITE;

--Refresh

ALTER MATERIALIZED VIEW <schema.materialized_view> REFRESH <FAST | COMPLETE | FORCE> ON <DEMAND | COMMIT> START WITH <date_time> NEXT <date_time> WITH PRIMARY KEY USING DEFAULT MASTER ROLLBACK SEGMENTUSING <ENFORCED | TRUSTED> CONSTRAINTS;

ALTER MATERIALIZED VIEW mv_complete REFRESH COMPLETE;

--Shrink

ALTER MATERIALIZED VIEW <schema.materialized_view> SHRINK SPACE <COMPACT | CASCADE>;

ALTER MATERIALIZED VIEW mv_simple ENABLE ROW MOVEMENT;

ALTER MATERIALIZED VIEW mv_simple SHRINK SPACE CASCADE;

--Table Compression

ALTER MATERIALIZED VIEW <schema.materialized_view> <COMPRESS | NOCOMPRESS>;

ALTER MATERIALIZED VIEW mv_simple COMPRESS;

--Table Partitioning

ALTER MATERIALIZED VIEW <schema.materialized_view> ....

--Dropping Materialized View

syntax

DROP MATERIALIZED VIEW [schema.] materialized_view [PRESERVE TABLE] ;

example

DROP MATERIALIZED VIEW sales_sum_mv;

vahidusefzadeh@gmail.com

useful query

---list mview last refreshed on year or longer ago

```
SELECT  OWNER, MVIEW_NAME,  TO_CHAR(LAST_REFRESH_DATE,'DD-MM-YY') LAST_REFRESHED_DATE FROM
DBA_MVIEWS WHERE LAST_REFRESH_DATE < SYSDATE-365 ORDER BY LAST_REFRESH_DATE ASC ;
```

---determine the master table

```
select OWNER, NAME, MASTER_OWNER, MASTER, LAST_REFRESH  from DBA_SNAPSHOT_REFRESH_TIMES ;
```

---FAST_REFRESHABLE mviews which couldn't make FAST refresh

```
SELECT  OWNER, MVIEW_NAME, REFRESH_METHOD, FAST_REFRESHABLE,   LAST_REFRESH_TYPE,
LAST_REFRESH_DATE, STALENESS, AFTER_FAST_REFRESH, COMPILE_STATE, STALE_SINCE  FROM DBA_MVIEWS
WHERE FAST_REFRESHABLE<>'NO' and LAST_REFRESH_TYPE<>'FAST' ORDER BY OWNER, MVIEW_NAME ;
```

---all MATERIALIZED VIEW LOGS

```
SELECT  LOG_OWNER, MASTER, LOG_TABLE, LOG_TRIGGER, ROWIDS,PRIMARY_KEY, OBJECT_ID,
FILTER_COLUMNS,SEQUENCE, INCLUDE_NEW_VALUES FROM DBA_MVIEW_LOGS ORDER BY  LOG_OWNER,
MASTER;
```

--- Determine if a Specific MVIEW is Being Refreshed

```
select o.owner, o.object_name mview, username, s.sid from v$lock l, dba_objects o, v$session s where
o.object_id=l.id1 and l.type='JI' and l.lmode=6   and s.sid=l.sid and o.object_type='TABLE';
```

User

-- CREATE USER

syntax

```
CREATE USER user   IDENTIFIED { BY password    | EXTERNALLY [ AS 'certificate_DN' | AS 'kerberos_principal_name'
]   | GLOBALLY [ AS '[ directory_DN ]' ]   }   [ DEFAULT TABLESPACE tablespace   | TEMPORARY TABLESPACE
      { tablespace | tablespace_group_name }   | { QUOTA { size_clause | UNLIMITED } ON tablespace }...
```

vahidusefzadeh@gmail.com

| PROFILE profile | PASSWORD EXPIRE | ACCOUNT { LOCK | UNLOCK } [DEFAULT TABLESPACE tablespace

| TEMPORARY TABLESPACE { tablespace | tablespace_group_name } | { QUOTA { size_clause |

UNLIMITED } ON tablespace }... | PROFILE profile | PASSWORD EXPIRE | ACCOUNT { LOCK | UNLOCK } |

ENABLE EDITIONS | CONTAINER = { CURRENT | ALL }]...] ;

example

```
CREATE USER user1 IDENTIFIED BY urs1754  TEMPORARY TABLESPACE TEMPTBS01  DEFAULT TABLESPACE user1ts

QUOTA 500M ON user1ts  PROFILE 'SALES_RPOF';
```

--ALTER USER

syntax

```
ALTER USER  { user { IDENTIFIED  { BY password [ REPLACE old_password ]

    | EXTERNALLY [ AS 'certificate_DN' | AS 'kerberos_principal_name' ]  | GLOBALLY [ AS '[directory_DN]' ]  }

  | DEFAULT TABLESPACE tablespace  | TEMPORARY TABLESPACE { tablespace | tablespace_group_name }

  | { QUOTA { size_clause  | UNLIMITED  } ON tablespace  } ... | PROFILE profile

  | DEFAULT ROLE { role [, role ]... | ALL [ EXCEPT role [, role ]... ]  | NONE  }  | PASSWORD EXPIRE

  | ACCOUNT { LOCK | UNLOCK }  | ENABLE EDITIONS [ FOR object_type [, object_type ]... ] [ FORCE ]

  | CONTAINER = { CURRENT | ALL }  | container_data_clause  } ...  | user [, user ]... proxy_clause } ;
```

--grant to users

sytax

```
GRANT  { { grant_system_privileges | grant_object_privileges }  [ CONTAINER = { CURRENT | ALL } ] }  |
grant_roles_to_programs ;
```

example

```
GRANT CREATE SESSION TO salapati;

ALTER USER user1 QUOTA 100M ON user1ts;

GRANT UNLIMITED TABLESPACE TO user1;
```

-- REVOKE from users

vahidusefzadeh@gmail.com

syntax

REVOKE { { revoke_system_privileges | revoke_object_privileges } [CONTAINER = { CURRENT | ALL }] } | revoke_roles_from_programs ;

example

REVOKE CREATE SESSION FROM user1;

-- UNLOCK/LOCK

ALTER USER usef ACCOUNT UNLOCK;

ALTER USER usef ACCOUNT LOCK;

-- dropping user

DROP USER user1;

DROP USER user1 cascade;

--useful query:

---user qoutas

```
SELECT USERNAME,TABLESPACE_NAME, BYTES/1024 SIZE_KB,  DECODE(MAX_BYTES,-1,-1,MAX_BYTES/1024/1024) MAX_MB  FROM DBA_TS_QUOTAS ORDER BY USERNAME;
```

---diskspace usage by USER

```
select owner, round(sum(bytes)/1024/1024,2) space_in_mb  from dba_segments l   where l.tablespace_name not IN('SYSTEM','SYSAUX')  group by owner order by round(sum(bytes)/1024/1024,2) desc;
```

---user temporary usage

```
SELECT USERNAME,SESSION_NUM SESSION_SN, SQLADDR,SQLHASH,SQL_ID, TABLESPACE, SEGTYPE,SEGFILE# INIT_EXTENT_FILE#,SEGBLK# INIT_EXTENT_BLK#,EXTENTS,BLOCKS,SEGRFNO# FROM V$TEMPSEG_USAGE;
```

---last modify password

```
select name, round(sysdate-ptime)  last_modify from sys.user$ where ltime is null order by name;
```

-- Creating and Using User Profiles

vahidusefzadeh@gmail.com

syntax

```
CREATE PROFILE profile   LIMIT { resource_parameters  | password_parameters }...;
```

example

```
CREATE PROFILE SALES_PROF  LIMIT  connect_time 120  failed_login_attempts 3
 idle_time 60  sessions_per_user 2;
```

--altering User Profiles

syntax

```
ALTER PROFILE profile LIMIT  { resource_parameters | password_parameters } ...  ;
```

example

```
ALTER PROFILE test  LIMIT  sessions_per_user 4  failed_login_attempts 4;
ALTER SYSTEM SET resource_limit=true;
 ALTER USER salapati PROFILE SALES_PROF;
 SELECT profile FROM dba_users WHERE username = 'USER1';
 SELECT DISTINCT resource_name, limit FROM dba_profiles WHERE profile='DEFAULT';
ALTER PROFILE DEFAULT   LIMIT  PASSWORD_LIFE_TIME 180  PASSWORD_GRACE_TIME 7
PASSWORD_REUSE_TIME UNLIMITED  PASSWORD_REUSE_MAX UNLIMITED  FAILED_LOGIN_ATTEMPTS 10
PASSWORD_LOCK_TIME 1  PASSWORD_VERIFY_FUNCTION verify_function_11G;
```

-- Dropping a User Profile

syntax

```
DROP PROFILE profile [ CASCADE ] ;
```

example

```
DROP PROFILE test CASCADE;
```

--Managin Passwords

```
select value from v$parameter where name='sec_case_sensitive_logon';
```

```
alter system set sec_case_sensitive_logon=false ;

 SELECT username, password, password_versions FROM dba_users order by 1;
```

--making a password expired

```
ALTER USER hr IDENTIFIED BY hr PASSWORD EXPIRE;

ALTER PROFILE test_profile  LIMIT PASSWORD_LIFE_TIME 30;  -- in days (refere to profile section)

ALTER USER hr PROFILE test_profile;
```

--password file

```
select value from v$parameter where upper(name)='REMOTE_LOGIN_PASSWORDFILE';

SELECT * FROM v$pwfile_users;
```

orapwd syntax

```
orapwd file=<fname> entries=<users> force=<y/n> asm=<y/n>   dbuniquename=<dbname> format=<legacy/12>
sysbackup=<y/n> sysdg=<y/n>  syskm=<y/n> delete=<y/n> input_file=<input-fname>

 orapwd FILE=testpwd PASSWORD=remorse1 ENTRIES=20
```

--other examples

```
orapwd delete=y password="N0Access!" dbuniquename=orabase



orapwd delete=y file=c:\app\oracle\product\12.1.0\dbhome_1\bin\PWDorabase.ora
```

--External (OS) Authentication

```
REMOTE_OS_AUTHENT=TRUE
```

-- create the OS user

```
useradd usef

passwd usef
```

--set the parameter OS_AUTHENT_PREFIX

```
SHOW PARAMETER os_authent_prefix

alter system set os_authent_prefix='ops$' scope=spfile;
```

```
alter system set OS_AUTHENT_PREFIX = '' scope=spfile;

 CREATE USER ops$usef  IDENTIFIED EXTERNALLY;

GRANT CONNECT TO ops$usef;

su - usef

sqlplus /
```

--CREATE ROLE

syntax

```
CREATE ROLE role   [ NOT IDENTIFIED   | IDENTIFIED { BY password | USING [ schema. ] package | EXTERNALLY
| GLOBALLY   } ] [ CONTAINER = { CURRENT | ALL } ];
```

example

```
CREATE ROLE new_dba;

CREATE ROLE clerk IDENTIFIED BY password;

CREATE ROLE admin_role IDENTIFIED USING usef.admin;

CREATE ROLE accts_rec IDENTIFIED EXTERNALLY;

CREATE ROLE supervisor IDENTIFIED GLOBALLY;
```

--ALTER ROLE

```
ALTER ROLE role  { NOT IDENTIFIED  | IDENTIFIED     { BY password      | USING [ schema. ] package | EXTERNALLY
| GLOBALLY     } } ;
```

example

```
ALTER ROLE warehouse_user NOT IDENTIFIED;

ALTER ROLE dw_manager IDENTIFIED BY data;

ALTER ROLE dw_manager IDENTIFIED USING hr.admin;
```

-- disable role

```
INSERT INTO SYSTEM.PRODUCT_USER_PROFILE(PRODUCT,userid,attribute,char_value)

VALUES('SQL*Plus','TESTER','ROLES','TEST123');
```

vahidusefzadeh@gmail.com

-- enable role

```
DELETE FROM product_user_profile  WHERE userid='TESTER'   AND char_value = 'TEST123';
```

-- Dropping a Role

```
DROP ROLE admin_user;
```

Database Link

--Privileges required

```
grant CREATE DATABASE LINK to hr;

 grant CREATE PUBLIC DATABASE LINK to hr;
```

-- CREATE DATABASE LINK

syntax

```
CREATE [ SHARED ] [ PUBLIC ] DATABASE LINK dblink  [ CONNECT TO

   { CURRENT_USER    | user IDENTIFIED BY password [ dblink_authentication ]    }

  | dblink_authentication  ]...  [ USING connect_string ] ;
```

example

```
 CREATE DATABASE LINK MONITOR CONNECT TO hr IDENTIFIED BY hr USING 'monitor';

 Create database link usefdb2.net connect to sa identified by a using '(DESCRIPTION=(ADDRESS = (PROTOCOL =
TCP)(HOST =10.4.x.x) (PORT=1521)) (connect_data=(service_name=usefdb)))';

CREATE PUBLIC DATABASE LINK MONITOR  CONNECT TO hr IDENTIFIED BY hr USING 'monitor';
```

--ALTER DATABASE LINK

syntax

```
ALTER DATABASE LINK dblink  { CONNECT TO user IDENTIFIED BY password [ dblink_authentication ]  |
dblink_authentication  };
```

example

```
ALTER DATABASE LINK private_link CONNECT TO hr IDENTIFIED BY hr_new_password;
```

ALTER PUBLIC DATABASE LINK public_link CONNECT TO scott IDENTIFIED BY scott_new_password;

ALTER SHARED PUBLIC DATABASE LINK shared_pub_link CONNECT TO scott IDENTIFIED BY scott_new_password AUTHENTICATED BY hr IDENTIFIED BY hr_new_password;

ALTER SHARED DATABASE LINK shared_pub_link CONNECT TO scott IDENTIFIED BY scott_new_password;

--DROP DATABASE LINK

syntax

DROP [PUBLIC] DATABASE LINK dblink ;

--view

select OWNER, DB_LINK, USERNAME, HOST, CREATED from DBA_DB_LINKS;

CHARACTER SET

--changing character set

startup open restrict

alter system set aq_tm_processes=0 scope=both;

alter system set job_queue_processes =0 scope=both;

shutdown abort

startup open restrict

alter database character set internal_use AR8MSWIN1256;

or

alter database character set internal_use al32utf8;

ALTER SYSTEM DISABLE RESTRICTED SESSION

SELECT * FROM nls_database_parameters WHERE parameter IN ('NLS_CHARACTERSET','NLS_NCHAR_CHARACTERSET');

Flashback

-- Flashback Table

vahidusefzadeh@gmail.com

syntax

FLASHBACK TABLE [schema.] table [, [schema.] table]... TO { { { SCN | TIMESTAMP } expr | RESTORE POINT restore_point } [{ ENABLE | DISABLE } TRIGGERS] | BEFORE DROP [RENAME TO table] } ;

example

ALTER TABLE emp ENABLE ROW MOVEMENT;

SELECT current_scn from V$DATABASE;

select DBMS_FLASHBACK.GET_SYSTEM_CHANGE_NUMBER from dual ;

FLASHBACK TABLE emp TO SCN 5759290864;

OR

FLASHBACK TABLE emp TO TIMESTAMP TO_TIMESTAMP ('2015-01-30 07:00:00', 'YYYY-MM-DD HH24:MI:SS');

OR

FLASHBACK TABLE emp TO TIMESTAMP TO_TIMESTAMP ('2015-04-05 10:00:00', 'YYYY-MM-DD HH24:MI:SS') ENABLE TRIGGERS;

-- Flashback Drop

FLASHBACK TABLE table_name TO BEFORE DROP;

FLASHBACK TABLE "BIN$xTMPjHZ6SG+1xnDIaR9E+g==$0" TO BEFORE DROP;

FLASHBACK TABLE "BIN$xTMPjHZ6SG+1xnDIaR9E+g==$0" TO BEFORE DROP RENAME TO NEW_PERSONS;

-- Flashback Database

---Configuring Flashback Database

ARCHIVE LOG LIST

ALTER DATABASE FLASHBACK ON;

ALTER DATABASE OPEN;

-- Exepting a tablespace

ALTER TABLESPACE users FLASHBACK OFF;

ALTER TABLESPACE users FLASHBACK ON;

www.usefzadeh.com

vahidusefzadeh@gmail.com

-- Disabling Flashback Database

ALTER DATABASE FLASHBACK OFF;

-- Flashback db in action

syntax

FLASHBACK [STANDBY] DATABASE [database] { TO { { SCN | TIMESTAMP } expr | RESTORE POINT restore_point } | TO BEFORE { SCN | TIMESTAMP} expr | RESETLOGS } }

examples

STARTUP MOUNT;

FLASHBACK DATABASE TO SCN 5964663;

FLASHBACK DATABASE TO SEQUENCE 12345;

FLASHBACK DATABASE TO TIMESTAMP(SYSDATE -1/24);

FLASHBACK DATABASE TO SCN 5964663 TO BEFORE RESETLOGS;

ALTER DATABASE OPEN READ ONLY;

ALTER DATABASE OPEN RESETLOGS;

RECOVER DATABASE;

-- current scn

SELECT current_scn FROM V$DATABASE;

-- Restore Points

SELECT name, scn, storage_size, time, guarantee_flashback_database FROM v$restore_point;

-- CREATE RESTORE POINT

syntax

CREATE RESTORE POINT restore_point [AS OF {TIMESTAMP | SCN} expr] [PRESERVE | GUARANTEE FLASHBACK DATABASE];

example

CREATE RESTORE POINT rp_test;

-- creating a guaranteed restore point

CREATE RESTORE POINT test_guarantee GUARANTEE FLASHBACK DATABASE;

--DROP RESTORE POINT

DROP RESTORE POINT rp_test;

-- FLASHBACK:

FLASHBACK DATABASE TO RESTORE POINT test_guarantee;

-- flashback db status

SELECT flashback_on FROM v$database;

--Enabling and Disabling Recycle Bin

ALTER SYSTEM SET recyclebin = OFF SCOPE = SPFILE;

ALTER SYSTEM SET recyclebin = ON SCOPE = SPFILE;

SELECT object_name, original_name FROM dba_recyclebin WHERE owner = 'HR';

--Purging Objects in Recycle Bin

PURGE TABLE int_admin_emp;

PURGE TABLESPACE example;

PURGE TABLESPACE example USER oe;

PURGE RECYCLEBIN;

--Restoring Tables from Recycle Bin

FLASHBACK TABLE int_admin_emp TO BEFORE DROP RENAME TO int2_admin_emp;

SELECT object_name, original_name, createtime FROM recyclebin;

FLASHBACK TABLE "BIN$yrMKlZaVMhfgNAgAIMenRA==$0" TO BEFORE DROP;

--Restoring Dependent Objects

SELECT OBJECT_NAME, ORIGINAL_NAME, TYPE FROM RECYCLEBIN;

FLASHBACK TABLE JOB_HISTORY TO BEFORE DROP;

SELECT INDEX_NAME FROM USER_INDEXES WHERE TABLE_NAME = 'JOB_HISTORY';

ALTER INDEX "BIN$DBo9UChtZSbgQFeMiAdCcQ==$0" RENAME TO JHIST_JOB_IX;

ALTER INDEX "BIN$DBo9UChuZSbgQFeMiAdCcQ==$0" RENAME TO JHIST_EMPLOYEE_IX;

AUDIT

--CREATE AUDIT POLICY

syntax

CREATE AUDIT POLICY policy [privilege_audit_clause] [action_audit_clause] [role_audit_clause] [WHEN

'audit_condition' EVALUATE PER { STATEMENT | SESSION | INSTANCE }] [CONTAINER = { ALL | CURRENT }] ;

example

CREATE AUDIT POLICY table_pol PRIVILEGES CREATE ANY TABLE, DROP ANY TABLE;

ALTER AUDIT POLICY dml_pol ADD PRIVILEGES CREATE ANY TABLE, DROP ANY TABLE;

ALTER AUDIT POLICY java_pol ADD ACTIONS CREATE JAVA, ALTER JAVA, DROP JAVA;

ALTER AUDIT POLICY table_pol ADD ROLES dba;

ALTER AUDIT POLICY security_pol ADD PRIVILEGES CREATE ANY LIBRARY, DROP ANY LIBRARY ACTIONS DELETE on hr.employees, INSERT on hr.employees, UPDATE on hr.employees, ALL on hr.departments ROLES dba, connect;

ACTIONS DELETE on hr.employees, INSERT on hr.employees, UPDATE on hr.employees, ALL on hr.departments ROLES dba, connect;

--DROP AUDIT POLICY

DROP AUDIT POLICY policy ;

--AUDIT (Traditional Auditing)

syntax

AUDIT { audit_operation_clause [auditing_by_clause | IN SESSION CURRENT]

| audit_schema_object_clause | NETWORK | DIRECT_PATH LOAD [auditing_by_clause]

} [BY { SESSION | ACCESS }] [WHENEVER [NOT] SUCCESSFUL] [CONTAINER = { CURRENT | ALL }];

example

audit session ;

AUDIT SESSION BY usef;

vahidusefzadeh@gmail.com

```
AUDIT select table BY usef BY SESSION;

AUDIT DELETE ANY TABLE BY usef WHENEVER NOT SUCCESSFUL;

AUDIT UPDATE ANY TABLE;

AUDIT SELECT,INSERT,UPDATE,DELETE ON employees BY ACCESS WHENEVER SUCCESSFUL;

AUDIT ALL PRIVILEGES;
```

--NOAUDIT (Traditional Auditing)

syntax

```
NOAUDIT    { audit_operation_clause [ auditing_by_clause ]   | audit_schema_object_clause

  | NETWORK   | DIRECT_PATH LOAD [ auditing_by_clause ]   }   [ WHENEVER [ NOT ] SUCCESSFUL ]

  [ CONTAINER = { CURRENT | ALL } ] ;
```

example

```
NOAUDIT select table BY usef;

NOAUDIT ALL; /* turns off all statement auditing */

NOAUDIT ALL PRIVILEGES; /* turns off all privilege auditing */

NOAUDIT ALL ON DEFAULT; /* turns off all object auditing */
```

--AUDIT (Unified Auditing)

syntax

```
AUDIT  { POLICY policy    [ { BY user [, user]... } | { EXCEPT user [, user]... } ]    [ WHENEVER [ NOT ] SUCCESSFUL ]

 }  |  { CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]...

     [, CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]... ]...    [ BY user [, user]... ]  } ;
```

--NOAUDIT (Unified Auditing)

syntax

```
NOAUDIT  { POLICY policy   | CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]...

    [, CONTEXT NAMESPACE namespace ATTRIBUTES attribute [, attribute ]... ]...  }  [ BY user [, user]... ] ;
```

--Clear audit-Table

vahidusefzadeh@gmail.com

truncate table sys.aud$;

----useful query

select * from DBA_STMT_AUDIT_OPTS;

select OS_USERNAME, USERNAME, USERHOST, ACTION_NAME, to_char(LOGOFF_TIME,'dd-mm-yy hh24:mi:ss') LOGOFF, SESSION_CPU from DBA_AUDIT_SESSION;

select USERNAME, USERHOST, ACTION_NAME, OBJ_NAME from DBA_AUDIT_OBJECT

Parallel

syntax

{NOPARALLEL |PARALLEL (DEGREE {degree | DEFAULT} [INSTANCES {instances | DEFAULT}])}

--Parallel Query

ALTER SESSION DISABLE PARALLEL QUERY;

ALTER SESSION ENABLE PARALLEL QUERY;

ALTER SESSION FORCE PARALLEL QUERY PARALLEL 32;

alter table usef parallel (degree 10);

ALTER TABLE USEF NOPARALLEL;

SELECT /*+ PARALLEL(usef,4) */ COUNT(*) FROM usef;

SELECT /*+ PARALLEL (16) */ …

SELECT /*+ PARALLEL */ …

SELECT /*+ PARALLEL(TABLE1, 16) */

SELECT /*+ PARALLEL(TABLE1 16) */ …

SELECT /*+ PARALLEL_INDEX(TABLE1, 16) */ …

SELECT /*+ PARALLEL_INDEX(TABLE1 16) */ …

--useful query

select degree from user_tables where table_name = 'USEF';

---

vahidusefzadeh@gmail.com

--Parallel DML

alter session disable parallel dml;

alter session enable parallel dml;

ALTER SESSION FORCE PARALLEL DML PARALLEL 32;

INSERT /*+ PARALLEL (useftbl,4,1) */ INTO useftbl   SELECT * FROM emp;

ALTER TABLE table_name PARALLEL (DEGREE 1 INSTANCES 1);

--Parallel DDL

 alter session disable parallel ddl;

alter session enable parallel ddl;

 ALTER SESSION FORCE PARALLEL DDL PARALLEL 32;

create table big_emp parallel (degree 4) as select * from emp;

CREATE INDEX emp_ix ON emp (emp_id)   TABLESPACE ind   STORAGE (INITIAL 1M NEXT 1M PCTINCREASE 0 MAXEXTENTS 20)   PARALLEL (DEGREE 4);

--Parallel Recovery

RECOVER DATABASE PARALLEL (DEGREE d INSTANCES y);

RECOVER DATAFILE 'datafile_name' PARALLEL (DEGREE d);

RECOVER DATABASE PARALLEL (DEGREE DEFAULT);

RECOVER TABLESPACE tbs1, tbs2 PARALLEL (degree 4);

RECOVER DATABASE NOPARALLEL;

Large Objects

--BLOB CREATE

CREATE TABLE <table_name> (column_name data_type, column_name data_type,column_name data_type)

LOB (lob_name) STORE AS (TABLESPACE <tablespace_name> STORAGE (INITIAL <lob_size>)

[CHUNK] <LOGGING | NOLOGGING> TABLESPACE <tablespace_name>;

example

CREATE TABLE blobtab (recid NUMBER(5), blobcol BLOB) LOB (blobcol) STORE AS blobseg (TABLESPACE uwdata STORAGE (INITIAL 1M) CHUNK 4096 NOCACHE NOLOGGING) TABLESPACE uwdata;

--CLOB CREATE

<LOB_storage_clause> ::= LOB { (LOB_item [, LOB_item]...) STORE AS [SECUREFILE | BASICFILE] (LOB_storage_parameters) | (LOB_item) STORE AS [SECUREFILE | BASICFILE] { LOB_segname (LOB_storage_parameters) | LOB_segname | (LOB_storage_parameters)} } <LOB_storage_parameters> ::= { TABLESPACE tablespace | { LOB_parameters [storage_clause]} | storage_clause} [TABLESPACE tablespace| { LOB_parameters [storage_clause]}]... <LOB_parameters> ::= [<ENABLE | DISABLE> STORAGE IN ROW] [CHUNK <chunk_size>] [PCTVERSION <integer>] [RETENTION < MAX | MIN integer | AUTO | NONE>] [FREEPOOLS <integer>] [LOB_deduplicate_clause] [LOB_compression_clause] LOB_encryption_clause] [<CACHE | NOCACHE | CACHE READS>] [<LOGGING | NOLOGGING>

example

CREATE TABLE clobtab (recid NUMBER(5), clobcol CLOB) LOB (clobcol) STORE AS clobseg (TABLESPACE uwdata STORAGE (INITIAL 1M) CHUNK 4096 NOCACHE NOLOGGING) TABLESPACE uwdata;

--Cache Reads

CREATE TABLE cache_test (testlob BLOB) LOB (testlob) STORE AS (CACHE READS);

SELECT table_name, cache FROM user_lobs;

ALTER TABLE cache_test MODIFY LOB (testlob) (NOCACHE);

--Move LOB

ALTER TABLE <table_name> MOVE TABLESPACE <tablespace_name> LOB (<lob_column_name>) STORE AS <lob_segment_name> (TABLESPACE <tablespace_name>);

ALTER TABLE lobtab MOVE LOB (lobcol) STORE AS (TABLESPACE example DISABLE STORAGE IN ROW);

SELECT segment_name, tablespace_name FROM user_segments WHERE segment_name IN ('LOBTAB', 'LOBSEG');

--Shrink LOB Storage

ALTER TABLE <table_name> MODIFY LOB (<column_name>) (SHRINK SPACE CASCADE);

ALTER TABLE blobtab MODIFY LOB(blobcol) (SHRINK SPACE CASCADE);

--Drop LOB segment

ALTER TABLE <table_name> DROP COLUMN <column_name>;

ALTER TABLE lobtab DROP COLUMN lobcol;

Partitioning

--Hash Partitioned Table

```
CREATE TABLE hash_part (
prof_history_id  NUMBER(10),
person_id        NUMBER(10) NOT NULL,
organization_id  NUMBER(10) NOT NULL,
record_date      DATE NOT NULL,
prof_hist_comments VARCHAR2(2000))
PARTITION BY HASH (prof_history_id)
PARTITIONS 3
STORE IN (part1, part2, part3);
```

```
SELECT table_name, tablespace_name, partitioned FROM user_tables ORDER BY 3;
SELECT partition_name, tablespace_name  FROM user_tab_partitions;
```

--Interval-Numeric Range Partitioned Table

```
CREATE TABLE interval_part (
person_id   NUMBER(5) NOT NULL,
first_name  VARCHAR2(30),
last_name   VARCHAR2(30))
```

```
PARTITION BY RANGE (person_id)

INTERVAL (100) STORE IN (part1) (

PARTITION p1 VALUES LESS THAN (101))

TABLESPACE uwdata;


SELECT table_name, tablespace_name, partitioned FROM user_tables ORDER BY 3;

SELECT table_name, partition_name, tablespace_name, high_value FROM user_tab_partitions

ORDER BY 1, 2;

INSERT INTO interval_part(person_id, first_name, last_name) VALUES (100, 'Dan', 'John');

SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name =
'INTERVAL_PART';

INSERT INTO interval_part (person_id, first_name, last_name) VALUES (101, 'Heli', 'Helskyaho');

 SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name =
'INTERVAL_PART';

INSERT INTO interval_part (person_id, first_name, last_name) VALUES (567, 'Tara', 'Havemeyer');

SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name =
'INTERVAL_PART';
```

--Interval-Date Range Partitioned Table

```
CREATE TABLE interval_date (
person_id NUMBER(5) NOT NULL,
last_name VARCHAR2(30),
dob      DATE)
PARTITION BY RANGE (dob)
INTERVAL (NUMTOYMINTERVAL(1,'MONTH'))
STORE IN (part2, part4, uwdata) (
PARTITION p1 VALUES LESS THAN (TO_DATE('2008-03-15','YYYY-MM-DD')));


INSERT INTO interval_date (person_id, last_name, dob) VALUES (1, 'John', SYSDATE-365);
INSERT INTO interval_date (person_id, last_name, dob) VALUES (2, 'Lofstrom', SYSDATE-365);
INSERT INTO interval_date (person_id, last_name, dob) VALUES (3, 'Havemeyer', SYSDATE-200);
INSERT INTO interval_date (person_id, last_name, dob) VALUES (4, 'Catz', SYSDATE-60);
INSERT INTO interval_date (person_id, last_name, dob) VALUES (5, 'Ellison', SYSDATE+60);
SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name =
'INTERVAL_DATE';
```

vahidusefzadeh@gmail.com

--Interval-Interval Range Partitioned Table with new partitions created every six months

```
CREATE TABLE interval_interval (
program_id    NUMBER,
line_number   NUMBER,
order_date    DATE)
PARTITION BY RANGE (order_date)
INTERVAL (NUMTOYMINTERVAL(6,'MONTH'))
STORE IN (uwdata) (
PARTITION p2000 VALUES LESS THAN (TO_DATE('01-JAN-2025', 'DD-MON-YYYY')))
ENABLE ROW MOVEMENT;

INSERT INTO interval_interval SELECT program_id, line_number, order_Date FROM airplanes;
SELECT partition_name, high_value FROM user_tab_partitions WHERE table_name = 'INTERVAL_INTERVAL';
```

--Interval-Interval Range Partitioned Table with new partitions created every hour using OLTP compression

```
CREATE TABLE hourly_interval (
datetime  DATE,
some_data NUMBER)
PARTITION BY RANGE (datetime)
INTERVAL (NUMTODSINTERVAL(1,'HOUR'))
STORE IN (part1, part2, part3) (
PARTITION p1 VALUES LESS THAN (TO_DATE('01-JAN-2015 00:00:00', 'DD-MON-YYYY HH24:MI:SS')))
COMPRESS FOR OLTP;

INSERT INTO hourly_interval (datetime, some_data) VALUES (SYSDATE, 1);
INSERT INTO hourly_interval (datetime, some_data) VALUES (SYSDATE+(1/24), 1);
INSERT INTO hourly_interval (datetime, some_data) VALUES (SYSDATE+(1/24), 1);
INSERT INTO hourly_interval (datetime, some_data) VALUES (SYSDATE+(3/24), 1);
COMMIT;
exec dbms_stats.gather_table_stats(USER,'HOURLY_INTERVAL', CASCADE=>TRUE);

SELECT table_name, partition_name, high_value, compression, compress_for FROM user_tab_partitions
WHERE table_name = 'HOURLY_INTERVAL';
```

--List Partitioned Table

```
CREATE TABLE list_part(
deptno         NUMBER(10),
deptname       VARCHAR2(20),
quarterly_sales NUMBER(10,2),
state          VARCHAR2(2))
PARTITION BY LIST (state) (
```

```
PARTITION q1_northwest VALUES ('OR', 'WA') TABLESPACE part1,
PARTITION q1_southwest VALUES ('AZ', 'CA', 'NM') TABLESPACE part2,
PARTITION q1_northeast VALUES ('NY', 'VT', 'NJ') TABLESPACE part3,
PARTITION q1_southeast VALUES ('FL', 'GA') TABLESPACE part4,
PARTITION q1_northcent VALUES ('MN', 'WI') TABLESPACE part1,
PARTITION q1_southcent VALUES ('OK', 'TX') TABLESPACE part2);

SELECT table_name, tablespace_name, partitioned FROM user_tables;
SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions;

INSERT INTO list_part VALUES (10, 'A', 1000, 'OR');
INSERT INTO list_part VALUES (20, 'B', 1000, 'AZ');
INSERT INTO list_part VALUES (10, 'A', 1000, 'WA');
INSERT INTO list_part VALUES (20, 'B', 1000, 'WA');
INSERT INTO list_part VALUES (10, 'A', 1000, 'AZ');
INSERT INTO list_part VALUES (20, 'B', 1000, 'CA');
COMMIT;

SELECT * FROM list_part;
SELECT * FROM list_part PARTITION(q1_northwest);
```

--Range Partitioned Table - By Date

```
CREATE TABLE range_part (
prof_history_id NUMBER(10),
person_id       NUMBER(10) NOT NULL,
organization_id NUMBER(10) NOT NULL,
record_date     DATE NOT NULL)
PARTITION BY RANGE (record_date) (
PARTITION yrmin VALUES LESS THAN (TO_DATE('01-JAN-2013','DD-MON-YYYY'))
TABLESPACE part1,
PARTITION yr13 VALUES LESS THAN (TO_DATE('01-JAN-2014','DD-MON-YYYY'))
TABLESPACE part2,
PARTITION yr14 VALUES LESS THAN (TO_DATE('01-JAN-2015','DD-MON-YYYY'))
TABLESPACE part3,
PARTITION yrmax VALUES LESS THAN (MAXVALUE) TABLESPACE part4);

SELECT table_name, tablespace_name, partitioned FROM user_tables WHERE table_name = 'RANGE_PART';

SELECT partition_name PART_NAME, tablespace_name TBSP_NAME, high_value FROM user_tab_partitions
WHERE table_name = 'RANGE_PART' ORDER BY 2;

INSERT INTO range_part VALUES (1, 1, 1, SYSDATE-720);
INSERT INTO range_part VALUES (2, 2, 2, SYSDATE);
INSERT INTO range_part VALUES (3, 3, 3, SYSDATE+180);
INSERT INTO range_part VALUES (4, 4, 4, SYSDATE+720);
```

```
SELECT * FROM range_part;
SELECT * FROM range_part PARTITION(yrmin);
SELECT * FROM range_part PARTITION(yr13);
SELECT * FROM range_part PARTITION(yr14);
SELECT * FROM range_part PARTITION(yrmax);
```

--Range Partitioned Table

```
CREATE TABLE students (
student_id NUMBER(6),
student_fn VARCHAR2(25),
student_ln VARCHAR2(25),
PRIMARY KEY (student_id))
PARTITION BY RANGE (student_ln) (
PARTITION student_ae VALUES LESS THAN ('F%') TABLESPACE part1,
PARTITION student_fl VALUES LESS THAN ('M%') TABLESPACE part2,
PARTITION student_mr VALUES LESS THAN ('S%') TABLESPACE part3,
PARTITION student_sz VALUES LESS THAN (MAXVALUE) TABLESPACE part4);

SELECT table_name, tablespace_name, partitioned FROM user_tables ORDER BY 3,1;
SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions;
```

--Referential Partitioned Table

```
SELECT MIN(num_rows), MAX(num_rows) FROM all_tables WHERE num_rows IS NOT NULL;

CREATE TABLE ref_parent (
table_name   VARCHAR2(30),
order_date   DATE,
num_rows     NUMBER)
PARTITION BY RANGE(num_rows) (
PARTITION num_rows1 VALUES LESS THAN (100) TABLESPACE part1,
PARTITION num_rows2 VALUES LESS THAN (1000) TABLESPACE part2,
PARTITION num_rows3 VALUES LESS THAN (10000) TABLESPACE part3,
PARTITION num_rows4 VALUES LESS THAN (MAXVALUE) TABLESPACE part4);

ALTER TABLE ref_parent ADD CONSTRAINT pk_ref_parent PRIMARY KEY (table_name) USING INDEX;
SELECT table_name, tablespace_name, partitioned FROM user_tables WHERE table_name = 'REF_PARENT';
SELECT partition_name, tablespace_name FROM user_tab_partitions WHERE table_name = 'REF_PARENT';

CREATE TABLE ref_child (
table_name VARCHAR2(30) NOT NULL,
index_name VARCHAR2(30) NOT NULL,
CONSTRAINT fk_ref_child_parent
FOREIGN KEY(table_name) REFERENCES ref_parent(table_name))
```

```
PARTITION BY REFERENCE(fk_ref_child_parent);

SELECT table_name, partitioning_type, ref_ptn_constraint_name FROM user_part_tables WHERE table_name LIKE
'REF%';
```

--Partition by System

```
CREATE TABLE syst_part (
tx_id   NUMBER(5),
begdate DATE)
PARTITION BY SYSTEM (
PARTITION p1 TABLESPACE part1,
PARTITION p2 TABLESPACE part2,
PARTITION p3 TABLESPACE part3);

INSERT INTO syst_part VALUES (1, SYSDATE-10);
          *
ERROR at line 1:
ORA-14701: partition-extended name or bind variable must be used for DMLs on tables partitioned by the System
method

INSERT INTO syst_part PARTITION (p1) VALUES (1, SYSDATE-10);
INSERT INTO syst_part PARTITION (p2) VALUES (2, SYSDATE);
INSERT INTO syst_part PARTITION (p3) VALUES (3, SYSDATE+10);
SELECT * FROM syst_part PARTITION(p2);
```

--Partition by Virtual Column

```
CREATE TABLE json_orders (
tx_id    NUMBER(5),
tx_date  DATE,
jsondata VARCHAR2(4000),
site_id  AS (JSON_VALUE(jsondata, '$.siteId' RETURNING NUMBER)))
PARTITION BY RANGE (site_id) (
PARTITION p1 VALUES LESS THAN (10),
PARTITION p2 VALUES LESS THAN (20),
PARTITION pm VALUES LESS THAN (MAXVALUE));

SELECT table_name, tablespace_name, partitioned FROM user_tables ORDER BY 3;

SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name =
'JSON_ORDERS';
SELECT column_name, virtual_column, data_default FROM user_tab_cols WHERE table_name = 'JSON_ORDERS';


INSERT INTO json_orders (tx_id, tx_date, jsondata) VALUES (1, SYSDATE, '{"Seattle": 1, "siteId": 9}');
```

```
INSERT INTO json_orders (tx_id, tx_date, jsondata) VALUES (2, SYSDATE, '{"New York": 2, "siteId": 11}');
SELECT * FROM json_orders;
SELECT * FROM json_orders PARTITION(p1);
SELECT * FROM json_orders PARTITION(p2);
```

--Composite Partitioned Table - By Range And Hash

```
CREATE TABLE composite_rng_hash (
cust_id     NUMBER(10),
cust_name   VARCHAR2(25),
amount_sold NUMBER(10,2),
time_id     DATE)
PARTITION BY RANGE(time_id)
SUBPARTITION BY HASH(cust_id)
SUBPARTITION TEMPLATE(
SUBPARTITION sp1 TABLESPACE part1,
SUBPARTITION sp2 TABLESPACE part2,
SUBPARTITION sp3 TABLESPACE part3,
SUBPARTITION sp4 TABLESPACE part4) (
PARTITION sales_pre11
VALUES LESS THAN (TO_DATE('01/01/2011','DD/MM/YYYY')),
PARTITION sales_2011
VALUES LESS THAN(TO_DATE('01/01/2012','DD/MM/YYYY')),
PARTITION sales_2012
VALUES LESS THAN(TO_DATE('01/01/2013','DD/MM/YYYY')),
PARTITION sales_2013
VALUES LESS THAN(TO_DATE('01/01/2014','DD/MM/YYYY')),
PARTITION sales_2014
VALUES LESS THAN(TO_DATE('01/01/2015','DD/MM/YYYY')),
PARTITION sales_future
VALUES LESS THAN(MAXVALUE));

SELECT table_name, partitioned, secondary FROM user_tables ORDER BY 2,1;
SELECT table_name, partition_name, composite, subpartition_count SPC, high_value
FROM user_tab_partitions;
SELECT table_name, partition_name, subpartition_name, subpartition_position FROM user_tab_subpartitions;

SELECT subpartition_name, tablespace_name, high_bound FROM user_subpartition_templates;

CREATE DATABASE LINK pdborcl CONNECT TO sh IDENTIFIED BY sh USING 'PDBORCL';

INSERT INTO composite_rng_hash SELECT c.cust_id, c.cust_first_name || ' ' || c.cust_last_name, s.amount_sold,
s.time_id+5000 FROM sh.sales@orcl s, sh.customers@orcl c WHERE s.cust_id = c.cust_id AND rownum < 250001;

exec dbms_stats.gather_table_stats(USER, 'COMPOSITE_RNG_HASH', 'SALES_PRE11');
exec dbms_stats.gather_table_stats(USER, 'COMPOSITE_RNG_HASH', 'SALES_2011');
```

```
exec dbms_stats.gather_table_stats(USER, 'COMPOSITE_RNG_HASH', 'SALES_2012');
exec dbms_stats.gather_table_stats(USER, 'COMPOSITE_RNG_HASH', 'SALES_2013');
exec dbms_stats.gather_table_stats(USER, 'COMPOSITE_RNG_HASH', 'SALES_2014');
exec dbms_stats.gather_table_stats(USER, 'COMPOSITE_RNG_HASH', 'SALES_FUTURE');

SELECT table_name, partition_name, num_rows FROM user_tab_partitions;
SELECT table_name, partition_name, subpartition_name, num_rows FROM user_tab_subpartitions;

exec dbms_stats.gather_table_stats(USER, 'COMPOSITE_RNG_HASH', GRANULARITY=>'ALL');


SELECT table_name, partition_name, subpartition_name, num_rows FROM user_tab_subpartitions;
SELECT dbms_metadata.get_ddl('TABLE', 'COMPOSITE_RNG_HASH') FROM dual;
```

--Composite Partitioned Table - By Range And List

```
CREATE TABLE composite_rng_list (

cust_id     NUMBER(10),

cust_name   VARCHAR2(25),

cust_state  VARCHAR2(2),

time_id     DATE)

PARTITION BY RANGE(time_id)

SUBPARTITION BY LIST (cust_state)

SUBPARTITION TEMPLATE(

SUBPARTITION west VALUES ('OR', 'WA') TABLESPACE part1,

SUBPARTITION east VALUES ('NY', 'CT') TABLESPACE part2,

SUBPARTITION cent VALUES ('OK', 'TX') TABLESPACE part3) (

PARTITION per1 VALUES LESS THAN (TO_DATE('01/01/2000','DD/MM/YYYY')),

PARTITION per2 VALUES LESS THAN (TO_DATE('01/01/2010','DD/MM/YYYY')),

PARTITION per3 VALUES LESS THAN (TO_DATE('01/01/2020','DD/MM/YYYY')),

PARTITION future VALUES LESS THAN(MAXVALUE));


SELECT table_name, partition_name, composite, high_value FROM user_tab_partitions;

SELECT table_name, partition_name, subpartition_name, num_rowsFROM user_tab_subpartitions;
```

vahidusefzadeh@gmail.com

--Composite Partitioned Table - By Range And Range

```
CREATE TABLE composite_rng_rng (
cust_id     NUMBER(10),
cust_name   VARCHAR2(25),
cust_state  VARCHAR2(2),
time_id     DATE)
PARTITION BY RANGE(time_id)
SUBPARTITION BY RANGE (cust_id)
SUBPARTITION TEMPLATE(
SUBPARTITION original VALUES LESS THAN (1001) TABLESPACE part1,
SUBPARTITION acquired VALUES LESS THAN (8001) TABLESPACE part2,
SUBPARTITION recent VALUES LESS THAN (MAXVALUE) TABLESPACE part3) (
PARTITION per1 VALUES LESS THAN (TO_DATE('01/01/2000','DD/MM/YYYY')),
PARTITION per2 VALUES LESS THAN (TO_DATE('01/01/2010','DD/MM/YYYY')),
PARTITION per3 VALUES LESS THAN (TO_DATE('01/01/2020','DD/MM/YYYY')),
PARTITION future VALUES LESS THAN (MAXVALUE));


SELECT table_name, partition_name, composite, high_value FROM user_tab_partitions;
SELECT table_name, partition_name, subpartition_name, num_rows FROM user_tab_subpartitions;
```

--Composite Partitioned Table - By List And Hash

```
CREATE TABLE composite_list_hash (
cust_id     NUMBER(10),
cust_name   VARCHAR2(25),
cust_state  VARCHAR2(2),
time_id     DATE)
PARTITION BY LIST(cust_state)
SUBPARTITION BY HASH (cust_id)
SUBPARTITION TEMPLATE(
SUBPARTITION sp1 TABLESPACE part1,
```

```
SUBPARTITION sp2 TABLESPACE part2,
SUBPARTITION sp3 TABLESPACE part3,
SUBPARTITION sp4 TABLESPACE part4) (
PARTITION west VALUES ('OR', 'WA'),
PARTITION east VALUES ('NY', 'CT'),
PARTITION cent VALUES ('IL', 'MN'));
```

--Composite Partitioned Table - By List And List

```
CREATE TABLE composite_list_list (
cust_id     NUMBER(10),
cust_name   VARCHAR2(25),
cust_state  VARCHAR2(2),
time_id     DATE)
PARTITION BY LIST(cust_state)
SUBPARTITION BY LIST (cust_id)
SUBPARTITION TEMPLATE(
SUBPARTITION beg VALUES (1,3,5) TABLESPACE part1,
SUBPARTITION mid VALUES (2,4,6) TABLESPACE part2,
SUBPARTITION end VALUES (7,8,9,0) TABLESPACE part3) (
PARTITION west VALUES ('OR', 'WA'),
PARTITION east VALUES ('NY', 'CT'),
PARTITION cent VALUES ('IL', 'MN'));
```

-- Composite Partitioned Table - By List And Range

```
CREATE TABLE composite_list_rng (
cust_id     NUMBER(10),
cust_name   VARCHAR2(25),
cust_state  VARCHAR2(2),
time_id     DATE)
PARTITION BY LIST(cust_state)
SUBPARTITION BY RANGE (cust_id)
SUBPARTITION TEMPLATE(
SUBPARTITION original VALUES LESS THAN (1001) TABLESPACE part1,
SUBPARTITION acquired VALUES LESS THAN (8001) TABLESPACE part2,
SUBPARTITION recent VALUES LESS THAN (MAXVALUE) TABLESPACE part3) (
PARTITION west VALUES ('OR', 'WA'),
PARTITION east VALUES ('NY', 'CT'),
PARTITION cent VALUES ('IL', 'MN'));
```

Composite Interval Partition

```
CREATE TABLE t(
sequence_id      NUMBER,
reservation_date DATE,
```

```
location_code    VARCHAR2(5))
PARTITION BY RANGE (reservation_date)
INTERVAL (NUMTOYMINTERVAL(1, 'MONTH'))
SUBPARTITION BY LIST(location_code)
SUBPARTITION TEMPLATE (
SUBPARTITION spart01 VALUES ('USA'),
SUBPARTITION spart02 VALUES ('IND'),
SUBPARTITION spart03 VALUES ('GER')) (
PARTITION root VALUES LESS THAN (TO_DATE('01-JAN-2015','DD-MON-YYYY')));
```

--Alter Table For Partitions

--Moving Non-Composite Partitions

ALTER TABLE <table_name> MOVE PARTITION <partition_name> TABLESPACE <tablespace_name>;

SELECT table_name, partition_name, tablespace_name FROM user_tab_partitions;

ALTER TABLE hash_part MOVE PARTITION sys_p26 TABLESPACE uwdata;

ALTER TABLE list_part MOVE PARTITION q1_southcent TABLESPACE uwdata NOLOGGING;

ALTER TABLE range_part MOVE PARTITION yr0 TABLESPACE uwdata;

ALTER TABLE composite_rng_hash MOVE PARTITION sales_pre98 TABLESPACE uwdata;

SELECT table_name, partition_name, tablespace_name FROM user_tab_partitions;

--Moving Partition and Rebuild Global Index

ALTER TABLE <table_name> MOVE SUBPARTITION <subpartition_name> TABLESPACE <tablespace_name>

UPDATE INDEXES [(<index_name>(<partition_name>));

-

ALTER TABLE composite_rng_hash MOVE PARTITION sales_pre11 TABLESPACE uwdata UPDATE INDEXES;

--Moving Subpartitions

ALTER TABLE <table_name> MOVE SUBPARTITION <subpartition_name> TABLESPACE <tablespace_name>;

SELECT partition_name, subpartition_name, tablespace_name FROM user_tab_subpartitions WHERE TABLE_NAME = 'COMPOSITE_RNG_HASH';

ALTER TABLE composite_rng_hash MOVE SUBPARTITION sales_pre11_sp1 TABLESPACE uwdata PARALLEL (DEGREE 2);

SELECT partition_name, subpartition_name, tablespace_name FROM user_tab_subpartitions WHERE TABLE_NAME = 'COMPOSITE_RNG_HASH';

--Setting a new default tablespace

ALTER TABLE <table_owner>.<table_name> MODIFY DEFAULT ATTRIBUTES TABLESPACE <tablespace_name>;

SELECT DISTINCT 'ALTER TABLE ' || table_owner || '.' || table_name || ' MODIFY DEFAULT ATTRIBUTES TABLESPACE newtbs;'

--Merging List Subpartitions

ALTER TABLE <table_name> MERGE SUBPARTITIONS <subpartition_name> INTO SUBPARTITION <subpartition_name TABLESPACE <tablespace_name>;

ALTER TABLE composite_rng_hash MERGE SUBPARTITIONS sales_pre11_sp1, sales_pre11_sp2 INTO SUBPARTITION sales_pre11_sp12 TABLESPACE part1;

```
CREATE TABLE range_list (
cust_id     NUMBER(10),
channel_id  NUMBER(3),
amount_sold NUMBER(10,2),
time_id     DATE)
PARTITION BY RANGE(time_id)
SUBPARTITION BY LIST(channel_id)
SUBPARTITION TEMPLATE(
SUBPARTITION sp1 VALUES (2, 3) TABLESPACE part1,
SUBPARTITION sp2 VALUES (4, 5) TABLESPACE part2,
```

```
SUBPARTITION sp3 VALUES (6, 7) TABLESPACE part3,

SUBPARTITION sp4 VALUES (8, 9) TABLESPACE part4)

(PARTITION sp10

VALUES LESS THAN(TO_DATE('01/01/2011','DD/MM/YYYY')),

PARTITION s11

VALUES LESS THAN(TO_DATE('01/01/2012','DD/MM/YYYY')),

PARTITION s12

VALUES LESS THAN(TO_DATE('01/01/2013','DD/MM/YYYY')),

PARTITION s13

VALUES LESS THAN(TO_DATE('01/01/2014','DD/MM/YYYY')),

PARTITION s14

VALUES LESS THAN(TO_DATE('01/01/2015','DD/MM/YYYY')),

PARTITION pm

VALUES LESS THAN(MAXVALUE));
```

```
SELECT partition_name, subpartition_name, tablespace_name, high_value FROM user_tab_subpartitions WHERE table_name = 'RANGE_LIST';

ALTER TABLE range_list MERGE SUBPARTITIONS sp10_sp1, sp10_sp2 INTO SUBPARTITION spmin PARALLEL (DEGREE 2) TABLESPACE part1;

SELECT partition_name, subpartition_name, tablespace_name, high_value FROM user_tab_subpartitions WHERE table_name = 'RANGE_LIST';
```

--Change The Tablespace Name For A Future Partition

```
ALTER TABLE <table_name> MODIFY DEFAULT ATTRIBUTES FOR PARTITION <partition_name> TABLESPACE <tablespace_name>;
```

```
SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name = 'RANGE_LIST';

ALTER TABLE range_list MODIFY DEFAULT ATTRIBUTES FOR PARTITION s11 TABLESPACE part1;
```

SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name = 'RANGE_LIST';

SELECT partition_name, subpartition_name, tablespace_name FROM user_tab_subpartitions WHERE table_name = 'RANGE_LIST';

--Change The Tablespace Store In List For An Interval Partitioned Table

ALTER TABLE <table_name> SET STORE IN <tablespace_name_list>;

SELECT dbms_metadata.get_ddl('TABLE', 'HOURLY_INTERVAL') FROM dual;

ALTER TABLE hourly_interval SET STORE IN (UWDATA, EXAMPLE);

SELECT dbms_metadata.get_ddl('TABLE', 'HOURLY_INTERVAL') FROM dual;

ALTER TABLE hourly_interval SET STORE IN (UWDATA, EXAMPLE, USERS);

SELECT dbms_metadata.get_ddl('TABLE', 'HOURLY_INTERVAL') FROM dual;

--Modify A List Partitioned List

ALTER TABLE <table_name> MODIFY PARTITION <partition_name> ADD VALUES (<values_list>);

SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name = 'LIST_PART';

ALTER TABLE list_part MODIFY PARTITION q1_northcent ADD VALUES ('MI', 'OH');

SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name = 'LIST_PART';

--Drop Values From A List Partitioned List

ALTER TABLE <table_name> MODIFY PARTITION <partition_name> DROP VALUES (<values_list>);

ALTER TABLE list_part MODIFY PARTITION q1_southwest DROP VALUES ('NM');

SELECT partition_name, tablespace_name, high_value FROM user_tab_partitions WHERE table_name = 'LIST_PART';

--Convert a partition into a stand-alone table

ALTER TABLE <table_name> EXCHANGE PARTITION <partition_name> WITH TABLE <new_table_name> <including | excluding> INDEXES <with | without> VALIDATION EXCEPTIONS INTO <schema.table_name>;

SELECT table_name, partition_name, num_rows FROM user_tab_partitions WHERE table_name = 'LIST_PART';

CREATE TABLE q1_northwest AS SELECT * FROM list_part WHERE 1=2;

SELECT * FROM list_part;

SELECT * FROM list_part PARTITION(q1_northwest);

ALTER TABLE list_part EXCHANGE PARTITION q1_northwest WITH TABLE q1_northwest INCLUDING INDEXES WITHOUT VALIDATION EXCEPTIONS INTO USEF.problems;

SELECT * FROM q1_northwest;

SELECT * FROM list_part;

--Convert a stand-alone table into a partition

ALTER TABLE <table_name> EXCHANGE PARTITION <partition_name> WITH TABLE <table_name> [INCLUDING INDEXES <WITH | WITHOUT> VALIDATION];

```
CREATE TABLE range_part (
rid  NUMBER,
col1 VARCHAR2(10),
col2 VARCHAR2(100))
PARTITION BY RANGE(rid) (
partition p1 VALUES LESS THAN (1000),
partition p3 VALUES LESS THAN (3000),
partition pm VALUES LESS THAN (MAXVALUE));

CREATE TABLE new_part (
rid  NUMBER,
col1 VARCHAR2(10),
col2 VARCHAR2(100));
```

```
INSERT /*+ APPEND ORDERED FULL(s1) USE_NL(s2) */ INTO new_part SELECT 3000 + TRUNC((rownum-1)/500,6),
TO_CHAR(rownum), RPAD('x',100)

FROM sys.source$ s1, sys.source$ s2 WHERE rownum <= 100000;

SELECT COUNT(*) FROM range_part;

SELECT COUNT(*) FROM new_part;


SELECT table_name, partition_name, high_value FROM user_tab_partitions;

ALTER TABLE range_part EXCHANGE PARTITION pm WITH TABLE new_part;

DROP TABLE range_part PURGE;

DROP TABLE new_part PURGE;

-- recreate and populate tables

set timing on

ALTER TABLE range_part EXCHANGE PARTITION pm WITH TABLE new_part WITHOUT VALIDATION;

set timing off

-- add some realistic constraints

ALTER TABLE range_part  ADD CONSTRAINT pk_range_part PRIMARY KEY(rid)  USING INDEX LOCAL;

ALTER TABLE new_part ADD CONSTRAINT pk_new_part PRIMARY KEY(rid)  USING INDEX;

set timing on

ALTER TABLE range_part EXCHANGE PARTITION pm WITH TABLE new_part INCLUDING INDEXES WITHOUT
VALIDATION;

set timing off

-- repeat again but this time do the following before the exchange

ALTER TABLE range_part MODIFY PRIMARY KEY NOVALIDATE;

ALTER TABLE new_part MODIFY PRIMARY KEY NOVALIDATE;

ALTER TABLE range_part EXCHANGE PARTITION pm WITH TABLE new_part INCLUDING INDEXES WITHOUT
VALIDATION;
```

vahidusefzadeh@gmail.com

--Partition Exchange with an Interval Partitioned Table

```
SQL> CREATE TABLE interval_date(
  2 per_id NUMBER(5) NOT NULL,
  3 lname  VARCHAR2(30),
  4 dob    DATE)
  5 PARTITION BY RANGE (dob)
  6 INTERVAL (NUMTOYMINTERVAL(1,'MONTH'))
  7 STORE IN (uwdata) (
  8 PARTITION p1 VALUES LESS THAN (TO_DATE('2014-01-01','YYYY-MM-DD')));

SQL> INSERT INTO interval_date (per_id, lname, dob) VALUES (0, 'Hurd', TO_DATE('31-DEC-2013'));

-- table to be added by exchange
SQL> CREATE TABLE interval_new AS SELECT * FROM interval_date WHERE 1=2;
SQL> INSERT INTO interval_new values (1, 'John', SYSDATE);
SQL> insert into interval_new values (2, 'Whalen', SYSDATE+1);
SQL> insert into interval_new values (3, 'Arlo', SYSDATE+2);
SQL>  SELECT partition_name PNAME, high_value  FROM user_tab_partitions WHERE table_name =
'INTERVAL_DATE';
PNAME      HIGH_VALUE
---------- --------------------------------------------------------------------------------
P1         TO_DATE(' 2014-01-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')

SQL> ALTER TABLE interval_date EXCHANGE PARTITION p2 WITH TABLE interval_new WITHOUT VALIDATION;
EXCHANGE PARTITION p2 WITH TABLE interval_new
*
ERROR at line 2:
```

ORA-02149: Specified partition does not exist

-- force creation of a partition with the correct high_value for the exchange

SQL> INSERT INTO interval_date VALUES (1, 'DUMMY', SYSDATE);

SQL> SELECT partition_name PNAME, high_value FROM user_tab_partitions WHERE table_name = 'INTERVAL_DATE';

PNAME   HIGH_VALUE

---------- --

P1   TO_DATE(' 2014-01-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')

SYS_P2614 TO_DATE(' 2014-06-01 00:00:00', 'SYYYY-MM-DD HH24:MI:SS', 'NLS_CALENDAR=GREGORIAN')

SQL> ALTER TABLE interval_date EXCHANGE PARTITION SYS_P2614 WITH TABLE interval_new WITHOUT VALIDATION;

SQL> SELECT *   FROM interval_date;

   PER_ID LNAME   DOB

---------- ------------------------------ --------------------

         0 Hurd   31-DEC-2013 00:00:00

         1 John   13-MAY-2014 22:24:12

         2 Whalen   14-MAY-2014 22:24:18

         3 Arlo   15-MAY-2014 22:24:24

--Rename a partition

ALTER TABLE <table_name> RENAME PARTITION <existing_partition_name> TO <new_partition_name>;

SELECT table_name, partition_name FROM user_tab_partitions;

ALTER TABLE range_list RENAME PARTITION sf TO sales_future;

SELECT table_name, partition_name FROM user_tab_partitions;

--Split Partition

ALTER TABLE <table_name> SPLIT PARTITION <partition_name> AT <range_definition> INTO (PARTITION <first_partition>, PARTITION <second_partition>) UPDATE GLOBAL INDEXES;

SELECT table_name, partition_name, high_value FROM user_tab_partitions WHERE table_name = 'RANGE_PART';

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('01-JAN-1998'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('01-JAN-1999'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('01-JAN-2000'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('01-JAN-2001'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('15-MAR-2001'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('16-SEP-2001'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('20-DEC-2001'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('01-JAN-2002'), 'A');

INSERT INTO range_part VALUES (1, 1, 1, TO_DATE('01-JAN-2003'), 'A');

COMMIT;

SELECT * FROM range_part;

SELECT * FROM range_part PARTITION(yr2a);

ALTER TABLE range_part SPLIT PARTITION yr2 AT (TO_DATE('30-JUN-2001','DD-MON-YYYY')) INTO (PARTITION yr2a, PARTITION yr2b) UPDATE GLOBAL INDEXES;

SELECT * FROM range_part PARTITION(yr2a);

SELECT table_name, partition_name, high_value FROM user_tab_partitions WHERE table_name = 'RANGE_PART';

--Split An LOB Partition

ALTER TABLE <table_name> SPLIT PARTITION <partition_name> AT <split location> INTO(PARTITION <new_partition_name> TABLESPACE <tablespace_name>" LOB <column_name> STORE AS (TABLESPACE <tablespace_name>), PARTITION <new_partition_name> LOB (<column_name>) STORE AS (TABLESPACE <tablespace_name>);

CREATE TYPE adheader_typ AS OBJECT (header_name   VARCHAR2(256), creation_date DATE, header_text VARCHAR2(1024), logo   BLOB);

/

vahidusefzadeh@gmail.com

```
CREATE TABLE print_media_part (
product_id       NUMBER(6),
ad_id            NUMBER(6),
ad_composite     BLOB,
ad_sourcetext    CLOB,
ad_finaltext     CLOB,
ad_fltextn       NCLOB,
ad_textdocs_ntab TEXTDOC_TAB,
ad_photo         BLOB,
ad_graphic       BFILE,
ad_header        ADHEADER_TYP)
NESTED TABLE ad_textdocs_ntab STORE AS textdoc_nt
PARTITION BY RANGE (product_id) (
PARTITION p1 VALUES LESS THAN (100),
PARTITION p2 VALUES LESS THAN (200));

ALTER TABLE print_media_part SPLIT PARTITION p2 AT (150) INTO ( PARTITION p2a TABLESPACE uwdata
LOB (ad_photo, ad_composite) STORE AS (TABLESPACE part1), PARTITION p2b
LOB (ad_photo, ad_composite) STORE AS (TABLESPACE part2));
```

--Coalesce Hash Partitions

```
ALTER TABLE <table_name> COALESCE PARTITION;
ALTER TABLE hash_part COALESCE PARTITION;
```

--Add Partition And Specify BLOB/LOB Storage

```
ALTER TABLE <table_name> ADD PARTITION <new_partition_name> VALUES LESS THAN (MAXVALUE)
```

LOB (<column_name>) STORE AS (TABLESPACE <tablespace_name);

ALTER TABLE print_media_part ADD PARTITION p3 VALUES LESS THAN (MAXVALUE) LOB (ad_photo, ad_composite) STORE AS (TABLESPACE part3) LOB (ad_sourcetext, ad_finaltext) STORE AS (TABLESPACE part4);

--Index Partitions

--Global Index Creation

CREATE INDEX <index_name>

ON <table_name> <column_name_list>;

SELECT i.index_name, i.composite, i.partition_name, i.high_value

FROM user_ind_partitions i, user_tab_partitions t

WHERE i.partition_name = t.partition_name

AND t.table_name = 'RANGE_PART';

SELECT partition_name FROM user_tab_partitions WHERE table_name = 'RANGE_PART';

CREATE INDEX gi_range_part_person_id ON range_part (person_id);

SELECT index_name, partitioned FROM user_indexes WHERE table_name = 'RANGE_PART';

DROP INDEX gi_range_part_person_id;

--Local Index Creation and Partition Pruning Demo

CREATE INDEX <index_name> ON <table_name> <column_name_list> LOCAL;

CREATE INDEX li_range_part_person_id ON range_part (person_id) LOCAL;

SELECT index_name, partitioned FROM user_indexes WHERE table_name = 'RANGE_PART';

SELECT ip.index_name, ip.composite, ip.partition_name, ip.high_value FROM user_ind_partitions ip, user_indexes ui WHERE ip.index_name = ui.index_name AND ui.table_name = 'RANGE_PART';

DROP INDEX li_range_part_person_id;

```
CREATE INDEX li_range_part_person_id

ON range_part (person_id)

LOCAL (

PARTITION yr0  TABLESPACE part1,

PARTITION yr1  TABLESPACE part2,

PARTITION yr2a TABLESPACE part3,

PARTITION yr2b TABLESPACE part4,

PARTITION yr9  TABLESPACE uwdata);


SELECT ip.index_name, ip.partition_name, ip.tablespace_name, ip.high_value

FROM user_ind_partitions ip, user_indexes ui WHERE ip.index_name = ui.index_name AND ui.table_name =
'RANGE_PART';

SELECT * FROM range_part;

SELECT * FROM range_part PARTITION(yr2a);

EXPLAIN PLAN FOR SELECT * FROM range_part WHERE record_date BETWEEN TO_DATE('01-JAN-1998') AND
TO_DATE('31-JAN-1998');

SELECT * FROM TABLE(dbms_xplan.display);

EXPLAIN PLAN FOR SELECT * FROM range_part WHERE record_date BETWEEN TO_DATE('01-JAN-1998') AND
TO_DATE('31-DEC-2000');

SELECT * FROM TABLE(dbms_xplan.display);

EXPLAIN PLAN FOR SELECT * FROM range_part WHERE record_date BETWEEN TO_DATE('01-JAN-1999') AND
TO_DATE('31-DEC-2002');

SELECT * FROM TABLE(dbms_xplan.display);
```

Global Partition Index Creation

```
CREATE INDEX <index_name>

ON <table_name> <column_name_list>

GLOBAL PARTITION BY RANGE (partition_column_name_list) (

PARTITION <partition_name> VALUES <condition>);
```

```
DROP INDEX li_range_part_person_id;


UPDATE range_part SET organization_id = ROWNUM;

SELECT * FROM range_part;


CREATE INDEX gi_range_part_person_id

ON range_part (organization_id)

GLOBAL PARTITION BY RANGE(organization_id) (

PARTITION p1 VALUES LESS THAN(4)

TABLESPACE part1,

PARTITION p2 VALUES LESS THAN(MAXVALUE)

TABLESPACE part2);


SELECT ip.index_name, ip.partition_name, ip.tablespace_name, ip.high_value FROM user_ind_partitions ip,
user_indexes ui WHERE ip.index_name = ui.index_name AND ui.table_name = 'RANGE_PART';
```

--Query for Unusable Indexes

```
SELECT index_name, partition_name, status FROM user_ind_partitions;
```

--Rebuild Local All Local Indexes On A Table

```
ALTER TABLE <table_name> MODIFY PARTITION <partition_name> REBUILD UNUSABLE LOCAL INDEXES;


CREATE INDEX li_range_part_person_id ON range_part (person_id) LOCAL;

SELECT t.table_name, i.index_name, i.partition_name, i.status FROM user_ind_partitions i, user_tab_partitions t

WHERE i.partition_name = t.partition_name;

ALTER TABLE range_part  MODIFY PARTITION yr0 REBUILD UNUSABLE LOCAL INDEXES;
```

--Rebuild any unusable local index partitions associated with a hash partition at the specific composite partitioned table subpartition level

vahidusefzadeh@gmail.com

```
ALTER TABLE <table_name>

MODIFY SUBPARTITION <subpartition_name>

REBUILD UNUSABLE LOCAL INDEXES;

SELECT i.table_name, s.index_name, s.partition_name, s.status

FROM user_ind_subpartitions s, user_indexes i

WHERE s.index_name = i.index_name;


ALTER TABLE composite_rng_hash MODIFY SUBPARTITION sales_1999_sp4 REBUILD UNUSABLE LOCAL INDEXES;
```

--Rebuild (and move) a local partition index

```
ALTER INDEX <index_name>

REBUILD PARTITION <partition_name>

TABLESPACE <new_tablespace_name>;

col partition_name format a10

col tablespace_name format a20


SELECT i.table_name, s.index_name, s.tablespace_name, s.partition_name, s.status FROM user_ind_partitions s,
user_indexes I WHERE s.index_name = i.index_name;

ALTER INDEX li_range_part_person_id  REBUILD PARTITION yr2 TABLESPACE uwdata;

SELECT i.table_name, s.index_name, s.tablespace_name, s.partition_name, s.status FROM user_ind_partitions s,
user_indexes I WHERE s.index_name = i.index_name;
```

--Setting a new default tablespace

```
ALTER INDEX <index_owner>.<index_name> MODIFY DEFAULT ATTRIBUTES TABLESPACE <tablespace_name>;


SELECT DISTINCT 'ALTER INDEX ' || index_owner || '.' || index_name ||  'MODIFY DEFAULT ATTRIBUTES
TABLESPACE newtbs;'
```

--Drop Partition

```
ALTER TABLE DROP PARTITION <partition_name> [UPDATE GLOBAL INDEXES];
```

vahidusefzadeh@gmail.com

SELECT table_name, partition_name FROM user_tab_partitions;

ALTER TABLE range_list DROP PARTITION s2k UPDATE GLOBAL INDEXES;

Autotrace

Syntax

SET AUTOT[RACE] {OFF | ON | TRACE[ONLY]} [EXP[LAIN]] [STAT[ISTICS]]

--Trace and Run

SQL> set autotrace on

SQL> SELECT * FROM dual;

--Trace Only

SQL> set autotrace traceonly

SQL> SELECT * FROM dual;

--Trace and Explain

SQL> set autotrace traceonly explain

SQL> SELECT * FROM dual;

--Trace and Statistics

SQL> set autotrace traceonly statistics

SQL> SELECT * FROM dual;

--Stop Tracing

set autotrace off

---

vahidusefzadeh@gmail.com

DBMS_REDEFINITION

--ABORT_REDEF_TABLE

Cleans Up Errors From The Redefinition Process.

```
dbms_redefinition.abort_redef_table(
uname      IN VARCHAR2,
orig_table IN VARCHAR2,
int_table  IN VARCHAR2,
part_name  IN VARCHAR2 := NULL);
```

example

```
exec dbms_redefinition.abort_redef_table('USEF', 'EMP','INT_EMP');
```

--CAN_REDEF_TABLE

Determines if a given table can be redefined online.

```
dbms_redefinition.can_redef_table(
uname       IN VARCHAR2,
tname       IN VARCHAR2,
options_flag IN BINARY_INTEGER := 1,
part_name    IN VARCHAR2       := NULL);
```

example

```
exec dbms_redefinition.can_redef_table('USEF', 'EMP', dbms_redefinition.cons_use_pk);
```

--COPY_TABLE_DEPENDENTS

Copies the dependant objects of the original table to the interim table.

```
dbms_redefinition.copy_table_dependents(
uname          IN  VARCHAR2,
orig_table     IN  VARCHAR2,
int_table      IN  VARCHAR2,
copy_indexes     IN  PLS_INTEGER := 1,
copy_triggers    IN  BOOLEAN     := TRUE,
copy_constraints IN  BOOLEAN     := TRUE,
copy_privileges  IN  BOOLEAN     := TRUE,
ignore_errors    IN  BOOLEAN     := FALSE,
num_errors       OUT PLS_INTEGER,
```

```
copy_statistics  IN  BOOLEAN     := FALSE
copy_mvlog       IN  BOOLEAN     := FALSE);
```

example

```
exec dbms_redefinition.copy_table_dependents('USEF', 'EMP', 'INT_EMP', 0, copy_constraints=>TRUE,
num_errors=>retval);
```

--FINISH_REDEF_TABLE

Registers a dependent object (index, trigger or constraint).

```
dbms_redefinition.finish_redef_table(
uname               IN VARCHAR2,
orig_table          IN VARCHAR2,
int_table           IN VARCHAR2,
part_name           IN VARCHAR2    := NULL
dml_lock_timeout    IN PLS_INTEGER := NULL,
continue_after_errors IN BOOLEAN   := FALSE);
```

example

```
exec dbms_redefinition.finish_redef_table('USEF', 'EMP', 'INT_EMP');
```

--REDEF_TABLE

Provides a single push-button interface that integrates several redefinition steps.

```
dbms_redefinition.redef_table(
uname                   IN VARCHAR2,
tname                   IN VARCHAR2,
table_compression_type    IN VARCHAR2 := NULL,
table_part_tablespace     IN VARCHAR2 := NULL,
index_key_compression_type IN VARCHAR2 := NULL,
index_tablespace          IN VARCHAR2 := NULL,
lob_compression_type      IN VARCHAR2 := NULL,
lob_tablespace            IN VARCHAR2 := NULL,
lob_store_as              IN VARCHAR2 := NULL);
```

example

```
exec dbms_redefinition.redef_table('USEF', 'SERVERS', 'COMPRESS FOR OLTP', 'SYSAUX');
```

--REGISTER_DEPENDENT_OBJECT

Completes The Redefinition Process.

```
dbms_redefinition.register_dependent_object(
uname        IN VARCHAR2,    -- schema name
orig_table   IN VARCHAR2,    -- table to redefine
int_table    IN VARCHAR2,    -- interim table
dep_type     IN PLS_INTEGER, -- type of dependent object
dep_owner    IN VARCHAR2,    -- owner of dependent object
dep_orig_name IN VARCHAR2,   -- name of orig dependent object
dep_int_name  IN VARCHAR2);  -- name of interim dependent obj.
```

example

```
exec dbms_redefinition.register_dependent_object('USEF', 'EMP', 'INT_EMP',dbms_redefinition.cons_trigger,
'USEF', 'bu1_hiredate', 'bu2_hiredate');
```

--START_REDEF_TABLE

Starts The Redefinition Process.

```
dbms_redefinition.start_redef_table(
uname              IN VARCHAR2,              -- schema name
orig_table         IN VARCHAR2,              -- table to redefine
int_table          IN VARCHAR2,              -- interim table
col_mapping        IN VARCHAR2      := NULL,    -- column mapping
options_flag       IN BINARY_INTEGER := 1,      -- redefinition type
orderby_cols       IN VARCHAR2      := NULL,    -- order by col list ASC/DESC
part_name          IN VARCHAR2      := NULL,    -- partition name
copy_vpd_opt       IN BINARY_INTEGER := 1,      -- by default do not copy
continue_after_errors IN BOOLEAN := FALSE);
```

example

```
exec dbms_redefinition.start_redef_table('USEF', 'EMP', 'INT_EMP', 'EMPNO EMPNO, ENAME NAME, SAL*1.10
SALARY, HIREDATE HIREDATE, DEPTNO DEPTNO', orderby_cols=>'EMPNO');
```

--SYNC_INTERIM_TABLE

Maintains Synchronization Between The Original And Interim Table.

```
dbms_redefinition.sync_interim_table(
uname              IN VARCHAR2,         -- schema name
orig_table         IN VARCHAR2,         -- original table
int_table          IN VARCHAR2,         -- interim table
part_name          IN VARCHAR2 := NULL,   -- partition name
continue_after_errors IN BOOLEAN  := FALSE);
```

example

exec dbms_redefinition.sync_interim_table('USEF', 'EMP', 'INT_EMP');

--UNREGISTER_DEPENDENT_OBJECT

Unregisters a dependent object.

dbms_redefinition.unregister_dependent_object(
uname   IN VARCHAR2,
orig_table   IN VARCHAR2,
int_table   IN VARCHAR2,
dep_type   IN PLS_INTEGER,
dep_owner   IN VARCHAR2,
dep_orig_name IN VARCHAR2,
dep_int_name IN VARCHAR2);

example

exec dbms_redefinition.unregister_dependent_object('USEF', 'EMP', 'INT_EMP', dbms_redefinition.cons_trigger, 'USEF', 'bu1_hiredate', 'bu2_hiredate');

External Tables

--Create Table

CREATE TABLE <table_name> (<column_definitions>) ORGANIZATION EXTERNAL (TYPE oracle_loader DEFAULT DIRECTORY <oracle_directory_object_name> ACCESS PARAMETERS (RECORDS DELIMITED BY newline BADFILE <file_name>DISCARDFILE <file_name> LOGFILE <file_name> [READSIZE <bytes>] [SKIP <number_of_rows> FIELDS TERMINATED BY '<terminator>' OPTIONALLY ENCLOSED BY '<character>' REJECT ROWS WITH ALL NULL FIELDS MISSING FIELD VALUES ARE NULL (<column_name_list>))\ LOCATION ('<file_name>')) [PARALLEL] REJECT LIMIT <UNLIMITED | integer>;

example 1

CREATE OR REPLACE DIRECTORY ext AS '/u01/external';

CREATE TABLE ext_tab1 (empno CHAR(4), ename CHAR(20), job   CHAR(20), deptno CHAR(3))

```
ORGANIZATION EXTERNAL (  TYPE oracle_loader  DEFAULT DIRECTORY ext  ACCESS PARAMETERS (
   RECORDS DELIMITED BY NEWLINE
   BADFILE ext:'bad_%a_%p.bad'
   LOGFILE ext:'log_%a_%p.log'
   FIELDS TERMINATED BY ','
   OPTIONALLY ENCLOSED BY '"'
   MISSING FIELD VALUES ARE NULL
   REJECT ROWS WITH ALL NULL FIELDS
   (empno, ename, job, deptno)) LOCATION ('demo1.dat')) PARALLEL REJECT LIMIT 0 NOMONITORING;

SELECT * FROM ext_tab;
SELECT table_name FROM user_tables;
SELECT table_name, type_name, default_directory_name,reject_limit, access_type FROM user_external_tables;
DROP TABLE ext_tab PURGE;
```

example 1

```
CREATE TABLE ext_tab2 (empno CHAR(4),ename CHAR(20),job CHAR(20),deptno CHAR(3))
ORGANIZATION EXTERNAL (  TYPE oracle_loader  DEFAULT DIRECTORY ext  ACCESS PARAMETERS (
   FIELDS TERMINATED BY ','
   MISSING FIELD VALUES ARE NULL
   (empno, ename, job, deptno))
  LOCATION ('demo1.dat','demo2.dat')) PARALLEL REJECT LIMIT 0;
SELECT * FROM ext_tab;
```

Alter Table

--Access Parameters

```
ALTER TABLE ext_tab2  ACCESS PARAMETERS (FIELDS TERMINATED BY '|');
```

vahidusefzadeh@gmail.com

--Add Column

ALTER TABLE ext_tab2 DROP COLUMN deptno;

--Default Directory

ALTER TABLE <table_name> DEFAULT DIRECTORY <directory_name>;

ALTER TABLE ext_tab2 DEFAULT DIRECTORY ctemp;

--Drop Column

ALTER TABLE <table_name> DROP COLUMN (<column_name>);

--Modify Column

ALTER TABLE <table_name> MODIFY (<column_name> <column_change>);

ALTER TABLE ext_tab2 MODIFY (deptno VARCHAR2(10));

--Parallel Access

ALTER TABLE <table_name> PARALLEL <integer>;

ALTER TABLE ext_tab2 PARALLEL 2;

--Project Column

ALTER TABLE <table_name> PROJECDT COLUMN <ALL | REFERENCED>;

ALTER TABLE ext_tab2 PROJECT COLUMN REFERENCED;

--Reject Limit

ALTER TABLE <table_name> REJECT LIMIT <integer>;

ALTER TABLE ext_tab2 REJECT LIMIT 2;

--Rename To

ALTER TABLE <current_table_name> RENAME TO <new_table_name>;

ALTER TABLE ext_tab2 RENAME TO ext_tab9;

--Set Unused

ALTER TABLE ext_tab LOCATION ('demo1.dat');

vahidusefzadeh@gmail.com

--Target File Name

ALTER TABLE <table_name> LOCATION ('<file1>', '<file2>', ...);

ALTER TABLE ext_tab LOCATION ('demo1.dat');

vahidusefzadeh@gmail.com

chapter 2

RAC

CRSCTL

Deprecated CRSCTL Commands and Replacements

Deprecated Command	Replacement Commands
crs_stat	crsctl check cluster crsctl status resource
crs_register	crsctl add resource crsctl add type crsctl modify resource crsctl modify type
crs_unregister	crsctl stop resource crsctl delete resource
crs_start	crsctl start resource crsctl start crs crsctl start cluster
crs_stop	crsctl stop resource crsctl stop crs crsctl stop cluster
crs_getperm	crsctl getperm resource crsctl getperm type
crs_profile	crsctl add resource crsctl add type crsctl status resource crsctl status type crsctl modify resource crsctl modify type
crs_relocate	crsctl relocate resource
crs_setperm	crsctl setperm resource crsctl setperm type
crsctl add crs administrator	Use the access control list (ACL) to control who can add server pools.
crsctl check crsd	crsctl check crs
crsctl check cssd	crsctl check css
crsctl check evmd	crsctl check evm
crsctl debug res log resource_name:level	crsctl set log
crsctl set css votedisk	crsctl add css votedisk crsctl delete css votedisk crsctl query css votedisk crsctl replace css votedisk

Deprecated Command	Replacement Commands
crsctl start resources	crsctl start resource -all
crsctl stop resources	crsctl stop resource -all

---

crsctl add resource

Use the crsctl add resource command to register a resource to be managed by Oracle Clusterware. A resource can be an application process, a database, a service, a listener, and so on.

Syntax

crsctl add resource resource_name -type resource_type [-file file_path | -attr "attribute_name=attribute_value ,attribute_name=attribute_value,..."] [-i] [-f]

Examples

Example 1

To register a VIP as a resource with Oracle Clusterware:

$ crsctl add resource app.appvip -type app.appvip.type -attr "RESTART_ATTEMPTS=2,

START_TIMEOUT=100, STOP_TIMEOUT=100, CHECK_INTERVAL=10,USR_ORA_VIP=172.16.0.0,

START_DEPENDENCIES=hard(ora.net1.network)pullup(ora.net1.network), STOP_DEPENDENCIES=hard(ora.net1.network)"

Example 2

To register a resource based on the test_type1 resource type:

$ crsctl add resource r1 -type test_type1 -attr "PATH_NAME=/tmp/r1.txt"

$ crsctl add resource r1 -type test_type1 -attr "PATH_NAME=/tmp/r2.txt"

Example 3

To register a Samba server resource of the generic_application resource type, using the EXECUTABLE_NAMES attribute:

```
# crsctl add resource my_samba -type generic_application -attr
"EXECUTABLE_NAMES=smbd,START_PROGRAM='/etc/rc.d/init.d/smb start',
STOP_PROGRAM='/etc/rc.d/init.d/smb stop'"
```

Example 4

To register a DNS server of the generic_application resource type, using the EXECUTABLE_NAMES attribute:

```
# crsctl add resource my_dns -type generic_application -attr
"EXECUTABLE_NAMES=named,START_PROGRAM='/etc/rc.d/init.d/named start',
STOP_PROGRAM='/etc/rc.d/init.d/named stop'"
```

Example 5

To register an Apache web server of the generic_application resource type using the PID_FILES attribute:

```
# crsctl add resource my_apache -type generic_application -attr "START_PROGRAM='/usr/sbin/httpd -k
start',STOP_PROGRAM='/usr/sbin/httpd -k stop', PID_FILES=/etc/httpd/run/httpd.pid"
```

Example 6

To register an application of generic_application resource type using environment variables:

```
# crsctl add resource my_app -type generic_application -attr "START_PROGRAM='/opt/my_app start',
EXECUTABLE_NAMES=my_app, ENVIRONMENT_VARS='USE_NETAPP=no ,USE_BACKUP=yes,CLEAN_ON_KILL=yes'"
```

---

crsctl add type

Use the crsctl add type command to create a resource type in Oracle Clusterware.

Syntax

```
crsctl add type type_name -basetype base_type_name {-attr "ATTRIBUTE=attribute_name | -file
file_path,TYPE={string | int}  [,DEFAULT_VALUE=default_value][,FLAGS=typeFlags"} [-i]
```

Example

```
# crsctl add type test_type1 -basetype cluster_resource   -attr "ATTRIBUTE=FOO,TYPE=integer,DEFAULT_VALUE=0"
```

vahidusefzadeh@gmail.com

---

crsctl add wallet

Use the crsctl add wallet command to create and add users to a wallet.

Syntax

crsctl add wallet -type wallet_type [-name name] [-user user_name -passwd]

Example

$ crsctl add wallet -type OSUSER -user lp_oracle_home_user -passwd

---

crsctl check css

Use the crsctl check css command to check the status of Cluster Synchronization Services. This command is most often used when Oracle Automatic Storage Management (Oracle ASM) is installed on the local server.

Syntax

crsctl check css

Example

CRS-4529: Cluster Synchronization Services is online

---

crsctl check evm

Use the crsctl check evm command to check the status of the Event Manager.

Syntax

crsctl check evm

Example

CRS-4533: Event Manager is online

vahidusefzadeh@gmail.com

---

crsctl delete resource

Use the crsctl delete resource command to remove resources from the Oracle Clusterware configuration.

Syntax

crsctl delete resource resource_name [-i] [-f]

Example

crsctl delete resource myResource

---

crsctl delete type

Use the crsctl delete type command to remove resource types from the Oracle Clusterware configuration.

Syntax

crsctl delete type type_name [-i]

Example

$ crsctl delete type test_type1 test_type2

---

crsctl delete wallet

Use the crsctl delete wallet command to remove wallets or users from a wallet.

Syntax

crsctl delete wallet -type wallet_type [-name name] [-user user_name]

Example

$ crsctl delete wallet -type OSUSER -user lp_oracle_home_user

vahidusefzadeh@gmail.com

---

crsctl eval add resource

Use the crsctl eval add resource command to predict the effects of adding a resource without making changes to the system. This command may be useful to application administrators.

Syntax

crsctl eval add resource resource_name -type resource_type [-attr

"attribute_name=attribute_value[,attribute_name=attribute_value[,..."]] | -file file_path] [-f]

---

crsctl eval fail resource

Use the crsctl eval fail resource command to predict the consequences of a resource failing.

Syntax

crsctl eval fail resource {resource_name | -w "filter"} [-n server]

Example

$ crsctl eval fail res cs1

---

crsctl eval modify resource

Use the crsctl eval modify resource command to predict the effects of modifying a resource without making changes to the system.

Syntax

crsctl eval modify resource resource_name -attr "attribute_name=attribute_value" [-f]

---

crsctl eval relocate resource

Use the crsctl eval relocate resource command to simulate relocating a resource without making changes to the system.

Syntax

crsctl eval relocate resource {resource_name | -all} {-s source_server |-w "filter"} [-n destination_server] [-f]

---

crsctl eval start resource

Use the crsctl eval start resource command to predict the effects of starting a resource without making changes to the system.

Syntax

crsctl eval start resource {resource_name [...] | -w "filter" | -all} [-n server_name] [-f]

---

crsctl eval stop resource

Use the crsctl eval stop resource command to predict the effects of stopping a resource without making changes to the system.

Syntax

crsctl eval stop resource {resource_name [...] | -w "filter" | -all} [-f]

---

crsctl get hostname

Use the crsctl get hostname command to retrieve the host name of the local server.

Syntax

crsctl get hostname

Example

$ crsctl get hostname

node2

vahidusefzadeh@gmail.com

---

crsctl getperm resource

Use the crsctl getperm resource command to display the user and group permissions for the specified resource.

Syntax

crsctl getperm resource resource_name [{-u user_name | -g group_name}]

Example

$ crsctl getperm resource app.appvip

Name: app.appvip

owner:root:rwx,pgrp:oinstall:rwx,other::r--

$ crsctl getperm resource app.appvip -u oracle

Name: app.appvip

rwx

$ crsctl getperm resource app.appvip -g dba

Name: app.appvip

r--

---

crsctl getperm type

Use the crsctl getperm type command to obtain permissions for a particular resource type.

Syntax

crsctl getperm type resource_type [-u user_name] | [-g group_name]

Example

$ crsctl getperm type app.appvip.type Name: app.appvip.type owner:root:rwx,pgrp:oinstall:rwx,other::r--

---

crsctl modify resource

Use the crsctl modify resource command to modify the attributes of a particular resource in Oracle Clusterware.

Syntax

crsctl modify resource resource_name -attr "attribute_name=attribute_value" [-i] [-f] [-delete]

Example

$ crsctl modify resource appsvip -attr USR_ORA_VIP=10.1.220.17 -i

---

crsctl modify type

Use the crsctl modify type command to modify an existing resource type.

Syntax

crsctl modify type type_name -attr "ATTRIBUTE=attribute_name,TYPE={string | int}
[,DEFAULT_VALUE=default_value [,FLAGS=[READONLY][| REQUIRED]]" [-i] [-f]]

Example

$ crsctl modify type myType.type -attr "ATTRIBUTE=FOO,DEFAULT_VALUE=0 ATTRIBUTE=BAR,
DEFAULT_VALUE=baz"

---

crsctl modify wallet

Use the crsctl modify wallet command to modify the password for a specific user in a specific wallet.

Syntax

crsctl modify wallet -type wallet_type [-name name] [-user user_name -passwd]

vahidusefzadeh@gmail.com

Example

$ crsctl modify wallet -type OSUSER -user lp_oracle_home_user -passwd

---

crsctl query wallet

Use the crsctl query wallet command to query low-privileged users from a wallet.

Syntax

crsctl query wallet -type wallet_type [-name name] [-user user_name] [-all]

Example

$ crsctl query wallet -type OSUSER -all

---

crsctl relocate resource

Use the crsctl relocate resource command to relocate resources to another server in the cluster.

Syntax

crsctl relocate resource {resource_name | resource_name | -all -s source_server |-w "filter"} [-n destination_server] [-k cid] [-env "env1=val1,env2=val2,..."] [-i] [-f]

Example

crsctl relocate resource myResource1 -s node1 -n node3

---

crsctl restart resource

Use the crsctl restart resource command to restart idle resources in the cluster, instead of having to run two commands to stop and start the resource.

Syntax

crsctl restart resource {resource_name [...] | -w "filter"} [-k cid] [-d did] [-env "env1=val1,env2=val2,..."] [-i] [-f]

Example

```
# crsctl restart resource myResource -s pool1 pool2
```

---

crsctl setperm resource

Use the crsctl setperm resource command to set permissions for a particular resource.

Example

```
$ crsctl setperm resource myResource -u user:scott:rwx
```

---

crsctl setperm type

Use the crsctl setperm type command to set permissions resources of a particular resource type.

Syntax

```
crsctl setperm type resource_type_name {-u acl_string | -x acl_string |-o user_name | -g group_name}
```

Example

```
$ crsctl setperm type resType -u user:scott:rwx
```

---

crsctl start resource

Use the crsctl start resource command to start many idle resources on a particular server in the cluster.

Syntax

```
crsctl start resource {resource_name [...] | -w "filter" | -all}   [-n server_name | -s server_pool_names] [-k cid] [-d did]   [-env "env1=val1,env2=val2,..."] [-begin] [-end] [-i] [-f] [-l]
```

Example

```
# crsctl start resource myResource -n server1
```

vahidusefzadeh@gmail.com

---

crsctl status resource

Use the crsctl status resource command to obtain the status and configuration information of many particular resources.

Syntax

To check the status of specific resources:

crsctl status resource resource_name [...] | -w "filter" [-p | -v] | [-f | -l | -g] [[-k cid | -n server_name] [-e [-p | -v]] [-d did]] | [-s -k cid [-d did]]

To print the status of the resources in tabular form:

crsctl status resource resource_name [...] | -w "filter" -t

To print a list of the resource dependencies:

crsctl status resource [resource_name [...]] -dependency [-stop | -pullup]

Examples

$ crsctl status resource ora.staii14.vip

---

crsctl status type

Use the crsctl status type command to obtain the configuration information of one or more particular resource types.

Syntax

crsctl status type [resource_type_name [...] | -w "filter"] [-g] [-p] [-f]

Example

$ crsctl status type ora.network.type

---

crsctl stop resource

Use the crsctl stop resource command to stop running resources.

Syntax

crsctl stop resource {resource_name [...] | -w "filter" | -all} [-n server_name] [-k cid] [-d did] [-env "env1=val1,env2=val2,..."] [-begin | -end] [-i] [-f] [-l]

Example

$ crsctl stop resource -n node1 -k 2

---

crsctl add category

Use the crsctl add category command to add a server category.

Syntax

crsctl add category category_name [-attr "attr_name=attr_value [,attr_name=attr_value[,...]]"] [-i]

Example

$ crsctl add category cat1 -attr "EXPRESSION='(CPU_COUNT > 2) AND (MEMORY_SIZE > 2048)'"

---

crsctl add crs administrator

Use the crsctl add crs administrator command to add a user to the list of cluster administrators.

Syntax

crsctl add crs administrator -u user_name [-f]

Example

crsctl add crs administrator -u scott

---

crsctl add css votedisk

Use the crsctl add css votedisk command to add one or more voting files to the cluster on storage devices other than an Oracle ASM disk group.

Syntax

crsctl add css votedisk path_to_voting_disk [path_to_voting_disk ...] [-purge]

Example

$ crsctl add css votedisk /stor/grid/ -purge

---

crsctl add policy

Use the crsctl add policy command to add a configuration policy to the policy set.

Syntax

crsctl add policy policy_name -attr "attr_name=attr_value[,attr_name=attr_value[, ...]]" [-i]

Example

$ crsctl add policy nightTime -attr "DESCRIPTION=nighttime"

---

crsctl add serverpool

Use the crsctl add serverpool command to add a server pool that is for hosting non-database resources (such as application servers) to Oracle Clusterware.

Syntax

crsctl add serverpool server_pool_name {-file file_path | -attr "attr_name=attr_value[,attr_name=attr_value[,...]]"} [-i] [-f]

Example 1

crsctl add serverpool testsp -attr "MAX_SIZE=5"

---

crsctl check cluster

Use the crsctl check cluster command on any node in the cluster to check the status of the Oracle Clusterware stack.

Syntax

crsctl check cluster [-all | [-n server_name [...]]

vahidusefzadeh@gmail.com

Example

$ crsctl check cluster -all

---

crsctl check crs

Use the crsctl check crs command to check the status of Oracle High Availability Services and the Oracle Clusterware stack on the local server.

Syntax

crsctl check crs

Example

$ crsctl check crs

CRS-4638: Oracle High Availablity Services is online

CRS-4537: Cluster Ready Services is onlin

CRS-4529: Cluster Synchronization Services is online

CRS-4533: Event Manager is online

---

crsctl check resource

Use the crsctl check resource command to initiate the check action inside the application-specific agent of a particular resource. Oracle Clusterware only provides output if something prevents the system from issuing the check request, such as a bad resource name.

Syntax

crsctl check resource {resource_name [...] | -w "filter" } [-n node_name] [-k cardinality_id] [-d degree_id] }

Example

$ crsctl check resource appsvip

vahidusefzadeh@gmail.com

---

crsctl check ctss

Use the crsctl check ctss command to check the status of the Cluster Time Synchronization services.

Syntax

crsctl check ctss

Example

CRS-4700: The Cluster Time Synchronization Service is in Observer mode.

or

CRS-4701: The Cluster Time Synchronization Service is in Active mode.

CRS-4702: Offset from the reference node (in msec): 100

---

crsctl config crs

Use the crsctl config crs command to display Oracle High Availability Services automatic startup configuration.

Syntax

crsctl config crs

Example

CRS-4622: Oracle High Availability Services autostart is enabled.

---

crsctl create policyset

Use the crsctl create policyset command to create a single policy set, in the form of a text file, that reflects the server pool configuration. After you create a policy set, you can copy the contents of the text file to create other policy sets.

Syntax

crsctl create policyset -file path_to_file

Example

$ crsctl create policyset -file /tmp/ps

---

crsctl delete category

Use the crsctl delete category command to delete a server category.

Syntax

crsctl delete category category_name [category_name [...]] [-i]

Example

$ crsctl delete category blue_server -i

---

crsctl delete crs administrator

Use the crsctl delete crs administrator command to remove a user from the Oracle Clusterware administrator list.

Syntax

crsctl delete crs administrator -u user_name [-f]

Example

crsctl delete crs administrator -u scott

---

crsctl delete css votedisk

Use the crsctl delete css votedisk to remove a voting file from the Oracle Clusterware configuration.

Syntax

crsctl delete css votedisk {voting_disk_GUID [...] | vdisk [...] | +diskgroup}

Example

$ crsctl delete css votedisk 26f7271ca8b34fd0bfcdc2031805581e

---

crsctl delete node

Use the crsctl delete node to remove a node from the cluster.

Syntax

crsctl delete node -n node_name

Example

crsctl delete node -n node06

---

crsctl delete policy

Use the crsctl delete policy command to delete a configuration policy from the policy set.

Syntax

crsctl delete policy policy_name [policy_name [...]] [-i]

---

crsctl delete serverpool

Use the crsctl delete serverpool command to remove a server pool from the Oracle Clusterware configuration.

Syntax

crsctl delete serverpool server_pool_name [server_pool_name [...]] [-i]

Example

crsctl delete serverpool sp1

vahidusefzadeh@gmail.com

---

crsctl disable crs

Use the crsctl disable crs command to prevent the automatic startup of Oracle High Availability Services when the server boots.

Syntax

crsctl disable crs

Example

CRS-4621: Oracle High Availability Services autostart is disabled.

---

crsctl discover dhcp

Use the crsctl discover dhcp command to send DHCP discover packets on the network at the specified port. If DHCP servers are present on the network, then they respond to the discovery message and the command succeeds.

Syntax

crsctl discover dhcp -clientid clientid [-port port]

Example

crsctl discover dhcp -clientid stmjk0462clr-stmjk01-vip

CRS-10009: DHCP server returned server: 192.168.53.232,

loan address : 192.168.29.221/255.255.252.0, lease time: 43200

---

crsctl enable crs

Use the crsctl enable crs command to enable automatic startup of Oracle High Availability Services when the server boots.

Syntax

crsctl enable crs

Example

CRS-4622: Oracle High Availability Services autostart is enabled.

---

crsctl eval activate policy

Use the crsctl eval activate policy command to predict the effects of activating a specific policy without making changes to the system. This command may be useful to cluster administrators.

Syntax

crsctl eval activate policy policy_name [-f] [-admin [-l serverpools | resources | all] [-x] [-a]]

---

crsctl eval add server

Use the crsctl eval add server command to simulate the addition of a server without making changes to the system.

Syntax

crsctl eval add server server_name [-file file_path] | [-attr "attr_name=attr_value[,...]"] [-admin [-l level [-x] [-a]] [-f]

Example

crsctl eval add server mjkeenan-node-3 -admin -l resources -a

---

crsctl eval add serverpool

Use the crsctl eval add serverpool command to predict the effects of adding a server pool without making changes to the system.

Syntax

crsctl eval add serverpool server_pool_name [-file file_path | -attr "attr_name=attr_value [,attr_name=attr_value[,...]]" [-admin [-l level [-x] [-a]] [-f]

vahidusefzadeh@gmail.com

---

crsctl eval delete server

Use the crsctl eval delete server command to predict the effects of deleting a server without making changes to the system.

Syntax

crsctl eval delete server server_name [-admin [-l level] [-x] [-a]] [-f]

---

crsctl eval delete serverpool

Use the crsctl eval delete serverpool command to simulate the deletion of a server pool without making changes to the system.

Syntax

crsctl eval delete serverpool server_pool_name [-admin [-l level] [-x] [-a]]

---

crsctl eval modify serverpool

Use the crsctl eval modify serverpool command to predict the effects of modifying a server pool without making changes to the system.

Syntax

crsctl eval modify serverpool server_pool_name {-file file_path | -attr "attr_name=attr_value [,attr_name=attr_value[, ...]]"} [-f] [-admin [-l level [-x] [-a]]

---

crsctl eval relocate server

Use the crsctl eval relocate server command to predict the effects of relocating a server to a different server pool without making changes to the system.

Syntax

crsctl eval relocate server server_name -to server_pool_name [-f][-admin [-l level] [-x] [-a]]

---

crsctl get clientid dhcp

Use the crsctl get clientid dhcp command to display the client ID that the Oracle Clusterware agent uses to obtain the IP addresses from the DHCP server for configured cluster resources. The VIP type is required.

Syntax

crsctl get clientid dhcp -cluname cluster_name -viptype vip_type [-vip vip_res_name] [-n node_name]

Example

$ crsctl get clientid dhcp -cluname stmjk0462clr -viptype HOSTVIP -n stmjk01

CRS-10007: stmjk0462clr-stmjk01-vip

---

crsctl get cluster hubsize

Use the crsctl get cluster hubsize command to obtain the value of Hub Nodes in an Oracle Flex Cluster.

Syntax

crsctl get cluster hubsize

Example

CRS-4950: Current hubsize parameter value is 32

---

crsctl get cluster mode

Use the crsctl get cluster mode command to ascertain whether the cluster is configured for Oracle Flex Clusters or the current status.

Syntax

crsctl get cluster mode [config | status]

---

crsctl get cpu equivalency

Use the crsctl cpu equivalency command to obtain the value of the CPU_EQUIVALENCY server configuration attribute.

Syntax

crsctl get cpu equivalency

---

crsctl get css

Use the crsctl get css command to obtain the value of a specific Cluster Synchronization Services parameter.

Syntax

crsctl get css parameter

Example

$ crsctl get css disktimeout

CRS-4678: Successful get disktimeout 200 for Cluster Synchronization Services.

---

crsctl get css ipmiaddr

Use the crsctl get css ipmiaddr command to get the address stored in the Oracle Local Registry of the local Intelligent Platform Management Interface (IPMI) device.

Syntax

crsctl get css ipmiaddr

Example

$ crsctl get css ipmiaddr

vahidusefzadeh@gmail.com

---

crsctl get css leafmisscount

Use the crsctl get css leafmisscount command to obtain the amount of time (in seconds) that must pass without any communication between a Leaf Node and the Hub Node to which it is attached, before the connection is declared to be no longer active and the Leaf Node is removed from the cluster.

Syntax

crsctl get css leafmisscount

---

crsctl get node role

Use the crsctl get node role command to obtain the configured node role of nodes in the cluster.

Syntax

crsctl get node role {config | status} [node node_name | -all]

Example

Node 'adc6140524' configured role is 'hub'

---

crsctl get nodename

Use the crsctl get nodename command to obtain the name of the local node.

Syntax

crsctl get nodename

Example

node2

---

crsctl get resource use

Use the crsctl get resource use command to check the current setting value of the RESOURCE_USE_ENABLED parameter.

vahidusefzadeh@gmail.com

Syntax

crsctl get resource use

Example

CRS-4966: Current resource use parameter value is 1

---

crsctl get server label

Use the crsctl get server label command to check the current setting value of the SERVER_LABEL server attribute.

Syntax

crsctl get server label

Example

CRS-4972: Current SERVER_LABEL parameter value is hubserver

---

crsctl getperm serverpool

Use the crsctl getperm serverpool command to obtain permissions for a particular server pool.

Syntax

crsctl getperm serverpool server_pool_name [-u user_name | -g group_name]

Example

$ crsctl getperm serverpool sp1

NAME: sp1

owner:root:rwx,pgrp:root:r-x,other::r--

---

crsctl lsmodules

Use the crsctl lsmodules command to list the components of the modules that you can debug.

vahidusefzadeh@gmail.com

Syntax

crsctl lsmodules {mdns | gpnp | css | crf | crs | ctss | evm | gipc}

Example

$ crsctl lsmodules evm

---

crsctl modify category

Use the crsctl modify category command to modify an existing server category.

Syntax

crsctl modify category category_name [-attr "attr_name=attr_value [,attr_name=attr_value[,...]]"] [-i] [-f]

Example

$ crsctl modify category blue_server -attr "EXPRESSION=(LOCATION=hub)"

---

crsctl modify policy

Use the crsctl modify policy command to modify an existing configuration policy.

Syntax

crsctl modify policy policy_name -attr "attr_name=attr_value" [-i]

Example

crsctl modify policy p1 -attr "DESCRIPTION=daytime"

---

crsctl modify policyset

Use the crsctl modify policyset command to modify an existing policy set.

Syntax

crsctl modify policyset {-attr "attr_name=attr_value[,attr_name=attr_value[, ...]]" | -file file_name} [-ksp]

Example

```
# crsctl modify policyset –file my_policy_set.def
```

---

crsctl modify serverpool

Use the crsctl modify serverpool command to modify an existing server pool.

Syntax

```
crsctl modify serverpool server_pool_name -attr "attr_name=attr_value   [,attr_name=attr_value[, ...]]" [-policy policyName | -all_policies][-i] [-f]
```

Example

```
# crsctl modify serverpool sp1 -attr "MAX_SIZE=7"
```

---

crsctl pin css

Use the crsctl pin css command to pin many specific nodes. Pinning a node means that the association of a node name with a node number is fixed. If a node is not pinned, its node number may change if the lease expires while it is down. The lease of a pinned node never expires.

Syntax

```
crsctl pin css -n node_name [ node_name [..]]
```

Example

```
# crsctl pin css -n node2
```

---

crsctl query crs administrator

Use the crsctl query crs administrator command to display the list of users with Oracle Clusterware administrative privileges.

Syntax

```
crsctl query crs administrator
```

Example

CRS Administrator List: scott

---

crsctl query crs activeversion

Use the crsctl query crs activeversion command to display the active version and the configured patch level of the Oracle Clusterware software running in the cluster. During a rolling upgrade, however, the active version is not advanced until the upgrade is finished across the cluster, until which time the cluster operates at the pre-upgrade version.

Syntax

crsctl query crs activeversion [-f]

Example

$ crsctl query crs activeversion -f

Oracle Clusterware active version on the cluster is [12.1.0.0.2]. The cluster

upgrade state is [NORMAL]. The cluster active patch level is [456789126].

---

crsctl query crs autostart

Use the crsctl query crs autostart command to obtain the values of the Oracle Clusterware automatic resource start criteria.

Syntax

crsctl query crs autostart

Example

'Autostart delay': 60

'Autostart servercount': 2

vahidusefzadeh@gmail.com

---

crsctl query crs releasepatch

Use the crsctl query crs releasepatch command to display the patch level which is updated in the Grid home patch repository while patching a node. The patch level corresponds to only the local node in which the command is executed. This command can be executed while the stack is not running.

Syntax

crsctl query crs releasepatch

Example

Oracle Clusterware release patch level is [3180840333] and the complete list of

---

crsctl query crs releaseversion

Use the crsctl query crs releaseversion command to display the version of the Oracle Clusterware software stored in the binaries on the local node.

Syntax

crsctl query crs releaseversion

Example

Oracle High Availablity Services release version on the local node is [11.2.0.2.0]

---

crsctl query crs softwarepatch

Use the crsctl query crs softwarepatch command to display the configured patch level of the installed Oracle Clusterware.

Syntax

crsctl query crs softwarepatch [host_name]

Example

Oracle Clusterware patch level on node [node1] is [456789126]

vahidusefzadeh@gmail.com

---

crsctl query crs softwareversion

Use the crsctl query crs softwareversion command to display latest version of the software that has been successfully started on the specified node.

Syntax

crsctl query crs softwareversion [node_name]

Example

Oracle Clusterware version on node [node1] is [11.2.0.2.0]

---

crsctl query css ipmiconfig

Use the crsctl query css ipmiconfig command to determine whether Oracle Clusterware on the local server has been configured to use IPMI for failure isolation. Note that this command detects the presence of configuration data, but cannot not validate its correctness.

Syntax

crsctl query css ipmiconfig

Example

CRS-4236: Oracle Clusterware configured to use IPMI

Or

CRS-4237: Oracle Clusterware is not fully configured to use IPMI

---

crsctl query css ipmidevice

Use the crsctl query css ipmiconfig command to determine the presence of the Intelligent Platform Management Interface (IPMI) driver on the local system.

Syntax

crsctl query css ipmidevice

Example

CRS-4231: IPMI device and/or driver found

Or

CRS-4218: Unable to access an IPMI device on this system

---

crsctl query css votedisk

Use the crsctl query css votedisk command to display the voting files used by Cluster Synchronization Services, the status of the voting files, and the location of the disks, whether they are stored on Oracle ASM or elsewhere.

Syntax

crsctl query css votedisk

Example

$ crsctl query css votedisk

---

crsctl query dns

Use the crsctl query dns command to obtain a list of addresses returned by DNS lookup of the name with the specified DNS server.

Syntax

crsctl query dns {-servers | -name name [-dnsserver DNS_server_address][-port port] [-attempts number_of_attempts] [-timeout timeout_in_seconds] [-v]}

Example

CRS-10024: DNS server returned 192.168.29.250 for name

stmjk07-vip.stmjk0462.foo.com

If you choose the -servers parameter, then the command returns output similar to the following:

CRS-10018: the following configuration was found on the system:

CRS-10019: There are 3 domains in search order. They are:

CRS-10022: There are 3 name servers. They are:

192.168.249.41

192.168.249.52

192.168.202.15

CRS-10020: number of retry attempts for name lookup is: 2

CRS-10021: timeout for each name lookup is: 1

---

crsctl query socket udp

Use the crsctl query socket udp command to verify that a daemon can listen on specified address and port.

Syntax

crsctl query socket udp [-address address] [-port port]

Examples

$ crsctl query socket udp

CRS-10030: could not verify if port 53 on local node is in use

crsctl query socket udp

CRS-10026: successfully created socket on port 53 on local node

The first of the preceding two commands was not run as root, and in both commands no port was specified, so CRSCTL assumed the default, 53, which is less than 1024. This condition necessitates running the command as root.

$ crsctl query socket udp -port 1023

CRS-10030: could not verify if port 1023 on local node is in use

crsctl query socket udp -port 1023

CRS-10026: successfully created socket on port 1023 on local node

Similar to the first two examples, the first of the preceding two commands was not run as root, and, although a port number was specified, it is still less than 1024, which requires root privileges to run the command.

In this last example, a port number greater than 1024 is specified, so there is no need to run the command as root:

$ crsctl query socket udp -port 1028

CRS-10026: successfully created socket on port 1028 on local node

---

crsctl release dhcp

Use the crsctl release dhcp command to send a DHCP lease release request to a specific client ID and send release packets on the network to a specific port.

Syntax

crsctl release dhcp -clientid clientid [-port port]

Example

$ crsctl release dhcp -clientid stmjk0462clr-stmjk01-vip

CRS-10012: released DHCP server lease for client ID stmjk0462clr-stmjk01-vip

on port 67

---

crsctl relocate resource

Use the crsctl relocate resource command to relocate resources to another server in the cluster.

Syntax

crsctl relocate resource {resource_name [-k cid] | {resource_name | -all} -s source_server | -w "filter"} [-n destination_server] [-env "env1=val1,env2=val2,..."] [-i] [-f]

Example

crsctl relocate resource myResource1 -s node1 -n node3

---

crsctl relocate server

Use the crsctl relocate server command to relocate a server to a different server pool.

vahidusefzadeh@gmail.com

Syntax

crsctl relocate server server_name [...] -c server_pool_name [-i] [-f]

Example

$ crsctl relocate server node6 node7 -c sp1

---

crsctl replace discoverystring

Use the crsctl replace discoverystring command to replace the existing discovery string used to locate voting files.

Syntax

crsctl replace discoverystring "absolute_path[,...]"

Example

crsctl replace discoverystring "/oracle/css1/*,/oracle/css2/*"

---

crsctl replace votedisk

Use the crsctl replace votedisk command to move or replace the existing voting files. This command creates voting files in the specified locations, either in Oracle ASM or some other storage option. Oracle Clusterware copies existing voting file information into the new locations and removes the voting files from the former locations.

Syntax

crsctl replace votedisk [+asm_disk_group | path_to_voting_disk [...]]

Example

Example 1

$ crsctl replace votedisk +diskgroup1

Example 2

$ crsctl replace votedisk /mnt/nfs/disk1 /mnt/nfs/disk2

vahidusefzadeh@gmail.com

---

crsctl request action

Use the crsctl request action command to perform a specific action on specific resource.

Syntax

crsctl request action action_name {-r resource_name [...] | -w "filter"} [-env "env1=val1,env2=val2,..."] [-i]

Example

$ crsctl request dhcp -clientid stmjk0462clr-stmjk01-vip

CRS-10009: DHCP server returned server: 192.168.53.232,

loan address : 192.168.29.228/255.255.252.0, lease time: 43200

---

crsctl request dhcp

Use the crsctl request dhcp command to send DHCP request packets on the network at the specified port. If the DHCP server has an IP address it can provide, then it responds with the IP address for the client ID.

Syntax

crsctl request dhcp -clientid clientid [-port port]

Example

$ crsctl request dhcp -clientid stmjk0462clr-stmjk01-vip

CRS-10009: DHCP server returned server: 192.168.53.232,

loan address : 192.168.29.228/255.255.252.0, lease time: 43200

---

crsctl set cluster hubsize

Use the crsctl set cluster hubsize command to set the maximum number of Hub Nodes for an Oracle Flex Cluster.

Syntax

crsctl set cluster hubsize

Example

$ crsctl set cluster hubsize 32

---

crsctl set cluster mode

Use the crsctl set cluster mode command to change a cluster to an Oracle Clusterware standard Cluster or an Oracle Flex Cluster.

Syntax

crsctl set cluster mode [standard | flex]

---

crsctl set cpu equivalency

Use the crsctl set cpu equivalency command to set a value for the CPU_EQUIVALENCY server configuration attribute.

Syntax

crsctl set cpu equivalency

---

crsctl set crs autostart

Use the crsctl set crs autostart command to set the Oracle Clusterware automatic resource start criteria. The autostart delay and minimum server count criteria delay Oracle Clusterware resource autostart until one of the two conditions are met.

Syntax

crsctl set crs autostart [delay delay_time] [servercount count]

Example

To ensure that Oracle Clusterware delays resource autostart for 60 seconds after the first server in the cluster is ONLINE:

crsctl set crs autostart delay 60

To ensure that Oracle Clusterware waits for there to be at least two servers ONLINE before it initiates resource autostart:

crsctl set crs autostart servercount 2

To ensure that Oracle Clusterware delays resource autostart until either of the previous two conditions are met (in no particular order):

crsctl set crs autostart delay 60 servercount 2

---

crsctl set css

Use the crsctl set css command to set the value of a Cluster Synchronization Services parameter.

Syntax

crsctl set css parameter value

---

crsctl set css ipmiaddr

Use the crsctl set css ipmiaddr command to store the address of the local Intelligent Platform Management Interface (IPMI) device in the Oracle Local Registry.

Syntax

crsctl set css ipmiaddr ip_address

Example

$ crsctl set css ipmiaddr 192.0.2.244

---

crsctl set css ipmiadmin

Use the crsctl set css ipmiadmin command to store the login credentials of an Intelligent Platform Management Interface (IPMI) administrator in the Oracle Local Registry.

Syntax

crsctl set css ipmiadmin ipmi_administrator_name

vahidusefzadeh@gmail.com

Example

$ crsctl set css ipmiadmin scott

---

crsctl set css leafmisscount

Use the crsctl set css leafmisscount command to specify, in seconds, the amount of time that must pass without any communication between a Leaf Node and the Hub Node to which it is attached, before the connection is declared to be no longer active and the Leaf Node is removed from the cluster.

Syntax

crsctl set css leafmisscount number_of_seconds

Example

$ crsctl set css leafmisscount 30

---

crsctl set node role

Use the crsctl set node role command to set the role of a specific node in the cluster.

Syntax

crsctl set node role [-node node_name] {hub | leaf}

Example

$ crsctl set node role -node node151 hub

---

crsctl set resource use

Use the crsctl set resource use command to set the value of the RESOURCE_USE_ENABLED server configuration parameter for the server on which you run this command.

Syntax

crsctl set resource use [1 | 0]

Example

crsctl set resource use 1

---

crsctl set server label

Use the crsctl set server label command to set the configuration value of the SERVER_LABEL server configuration attribute for the server on which you run this command.

Syntax

crsctl set server label value

Example

$ crsctl set server label hubserver

---

crsctl setperm serverpool

Use the crsctl setperm serverpool command to set permissions for a particular server pool.

Syntax

crsctl setperm serverpool server_pool_name {-u acl_string | -x acl_string |-o user_name | -g group_name}

Example

crsctl setperm serverpool sp3 -u user:jane.doe:rwx

---

crsctl start cluster

Use the crsctl start cluster command on any node in the cluster to start the Oracle Clusterware stack.

Syntax

crsctl start cluster [-all | -n server_name [...]]

Example

crsctl start cluster -n node1 node2

---

crsctl start crs

Use the crsctl start crs command to start Oracle High Availability Services on the local server.

Syntax

crsctl start crs [-excl [-nocrs] [-cssonly]] | [-wait | -waithas | -nowait] | [-noautostart]

Example

crsctl start crs

---

crsctl start ip

Use the crsctl start ip command to start a given IP name or IP address on a specified interface with a specified subnet mask. Run this command on the server on which you want to start the IP.

Syntax

crsctl start ip -A {IP_name | IP_address}/netmask/interface_name

Example

$ crsctl start ip -A 192.168.29.220/255.255.252.0/eth0

---

crsctl start rollingpatch

The crsctl start rollingpatch command transitions Oracle Clusterware and Oracle ASM into rolling patch mode. In this mode, the software tolerates nodes having different patch levels.

Syntax

crsctl start rollingpatch

vahidusefzadeh@gmail.com

---

crsctl start rollingupgrade

The crsctl start rollingupgrade command transitions Oracle Clusterware and Oracle ASM into rolling upgrade mode.

Syntax

crsctl start rollingupgrade version

---

crsctl start testdns

Use the crsctl start testdns command to start a test DNS server that will listen on a specified IP address and port. The test DNS server does not respond to incoming packets but does display the packets it receives. Typically, use this command to check if domain forwarding is set up correctly for the GNS domain.

Syntax

crsctl start testdns [-address address [-port port]] [-once] [-v]

Example

$ crsctl start testdns -address 192.168.29.218 -port 63 -v

---

crsctl status category

Use the crsctl status category command to obtain information about a server category.

Syntax

crsctl status category {category_name [category_name [...]] | [-w "filter" | -server server_name]}

Examples

$ crsctl stat category -w "ACTIVE_CSS_ROLE = hub"

NAME=my_category_i

ACL=owner:mjkeenan:rwx,pgrp:svrtech:rwx,other::r--

ACTIVE_CSS_ROLE = hub

EXPRESSION=(CPU_COUNT > 3)

To obtain the status of a server category by server:

$ crsctl stat category -server node1

NAME=my_category

ACL=owner:mjkeenan:rwx,pgrp:svrtech:rwx,other::r--

ACTIVE_CSS_ROLE = hub

EXPRESSION=

---

crsctl status ip

Use the crsctl status ip command to check if a given IP address is up on the network.

Syntax

crsctl status ip -A {IP_name | IP_address}

Example

CRS-10003: IP address 192.168.29.220 could be reached from current node

---

crsctl status policy

Use the crsctl status policy command to view the status and definition of a configuration policy.

Syntax

crsctl status policy [policy_name [policy_name [...]] | -w "filter" | -active]

---

crsctl status policyset

Use the crsctl status policyset command to view the current policies in the policy set, including the access control list, which governs who can modify the set, the last activated policy, and the configuration which is now in effect, which is known as the Current policy.

vahidusefzadeh@gmail.com

Syntax

crsctl status policyset [-file file_name]

---

crsctl status server

Use the crsctl status server command to obtain the status and configuration information of one or more particular servers.

Syntax

crsctl status server {server_name [...] | -w "filter"} [-g | -p | -v | -f] | [-category category_name | -w "filter"]

Example

$ crsctl status server node2 -f

NAME=node2

STATE=ONLINE

ACTIVE_POOLS=Generic ora.usefdb

STATE_DETAILS=

---

crsctl status serverpool

Use the crsctl status serverpool command to obtain the status and configuration information of one or more particular server pools.

Syntax

crsctl status serverpool [server_pool_name [...] | -w "filter"] [-p | -v | -f]crsctl status serverpool {[server_pool_name [...]} -g

Examples

Example 1

To display the full configuration of the server pool sp1:

$ crsctl status serverpool sp1 -f

NAME=spl

IMPORTANCE=1

MIN_SIZE=0

MAX_SIZE=-1

SERVER_NAMES=node3 node4 node5

PARENT_POOLS=Generic

EXCLUSIVE_POOLS=

ACL=owner:oracle:rwx,pgrp:oinstall:rwx,other::r--

ACTIVE_SERVERS=node3 node4

Example 2

To display all the server pools and the servers associated with them, use the following command:

$ crsctl status serverpool

NAME=Free

ACTIVE_SERVERS=

NAME=Generic

ACTIVE_SERVERS=node1 node2

NAME=ora.usefdb

ACTIVE_SERVERS=node1 node2

NAME=sp1

ACTIVE_SERVERS=node3 node4

Example 3

To find a server pool that meets certain criteria, use the following command:

$ crsctl status serverpool -w "MAX_SIZE > 1"

NAME=sp2

ACTIVE_SERVERS=node3 node4

vahidusefzadeh@gmail.com

---

crsctl status testdns

Use the crsctl status testdns command to query the test DNS server running on a specified address and local host name.

Syntax

crsctl status testdns [-server DNS_server_address] [-port port] [-v]

Example

CRS-10024: DNS server returned 192.168.28.74 for name

stmjk07-vip.stmjk0462.foo.com

---

crsctl stop cluster

Use the crsctl stop cluster command on any node in the cluster to stop the Oracle Clusterware stack on all servers in the cluster or specific servers.

Syntax

crsctl stop cluster [-all | -n server_name [...]] [-f]

Example

crsctl stop cluster -n node1

---

crsctl stop crs

Use the crsctl stop crs command to stop Oracle High Availability Services on the local server.

Syntax

crsctl stop crs [-f]

Example

crsctl stop crs

---

crsctl stop ip

Use the crsctl stop ip command to stop a given IP name or IP address on a specified interface with a specified subnet mask. Run this command on the server on which you want to stop the IP.

Syntax

crsctl stop ip -A {IP_name | IP_address}/interface_name

Example

$ crsctl stop ip -A MyIP.domain.com/eth0

---

crsctl stop rollingpatch

The crsctl stop rollingpatch command transitions Oracle Clusterware and Oracle ASM out of rolling patch mode. Once transitioned out of rolling patch mode, the software does not tolerate nodes having different patch levels.

Syntax

crsctl stop rollingpatch

---

crsctl stop testdns

Use the crsctl stop testdns command to stop a test DNS server.

Syntax

crsctl stop testdns [-address address [-port port]] [-domain GNS_domain] [-v]

Example

CRS-10032: successfully stopped the DNS listening daemon running on port 53 on

local node

vahidusefzadeh@gmail.com

---

crsctl unpin css

Use the crsctl unpin css command to unpin many servers. If a node is not pinned, its node number may change if the lease expires while it is down.

Syntax

crsctl unpin css -n node_name [node_name [...exit]]

Example

$ crsctl unpin css -n node1 node4

---

crsctl unset css

Use the crsctl unset css command to unset the value of a Cluster Synchronization Services parameter and restore it to its default value.

Syntax

crsctl unset css parameter

Example

$ crsctl unset css reboottime

---

crsctl unset css ipmiconfig

Use the crsctl unset css ipmiconfig command to clear all previously stored IPMI configuration (login credentials and IP address) from the Oracle Local Registry. This is appropriate when deconfiguring IPMI in your cluster or if IPMI configuration was previously stored by the wrong user.

Syntax

crsctl unset css ipmiconfig

Example

crsctl unset css ipmiconfig

crsctl stop crs

crsctl start crs

---

crsctl unset css leafmisscount

Use the crsctl unset css leafmisscount command to clear the amount of time that passes before the grace time begins after communication fails between a Hub Node and a Leaf Node and reset to the default.

Syntax

crsctl unset css leafmisscount

---

crsctl check has

Use the crsctl check has command to check the status of ohasd.

Syntax

crsctl check has

Example

CRS-4638: Oracle High Availability Services is online

---

crsctl config has

Use the crsctl check has command to display the automatic startup configuration of the Oracle High Availability Services stack on the server.

Syntax

crsctl config has

Example

CRS-4622 Oracle High Availability Services autostart is enabled.

---

vahidusefzadeh@gmail.com

crsctl disable has

Use the crsctl disable has command to disable automatic startup of the Oracle High Availability Services stack when the server boots up.

Syntax

crsctl disable has

Example

CRS-4621 Oracle High Availability Services autostart is disabled.

---

crsctl enable has

Use the crsctl enable has command to enable automatic startup of the Oracle High Availability Services stack when the server boots up.

Syntax

crsctl enable has

Example

CRS-4622 Oracle High Availability Services autostart is enabled.

---

crsctl query has releaseversion

Use the crsctl query has releaseversion command to display the release version of the Oracle Clusterware software that is stored in the binaries on the local node.

Syntax

crsctl query has releaseversion

Example

Oracle High Availability Services release version on the local node is [11.2.0.0.2]

---

crsctl query has softwareversion

Use the crsctl query has softwareversion command to display the software version on the local node.

Syntax

crsctl query has softwareversion

Example

Oracle High Availability Services version on the local node is [11.2.0.2.0]

---

crsctl start has

Use the crsctl start has command to start Oracle High Availability Services on the local server.

Syntax

crsctl start has

Example

crsctl start has

---

crsctl stop has

Use the crsctl stop has command to stop Oracle High Availability Services on the local server.

Syntax

crsctl stop has [-f]

Example

crsctl stop has

---

crsctl set log

Use the crsctl set log command to set log levels for Oracle Clusterware.

Syntax

crsctl set log {[crs | css | evm "component_name=log_level, [...]"] | [all=log_level]}

You can also set log levels for the agents of specific resources, as follows:

crsctl set log res "resource_name=log_level, [...]"

Examples

$ crsctl set log crs "CRSRTI=1,CRSCOMM=2"

To set log levels for all components of the EVM module:

$ crsctl set log evm all=2

To set a log level for a resource:

$ crsctl set log res "myResource1=3"

OCRCONFIG

---

ocrconfig -add

Use the ocrconfig -add command to add an OCR location to a shared file system or Oracle Automatic Storage Management (Oracle ASM) disk group. OCR locations that you add must exist, have sufficient permissions, and, in the case of Oracle ASM disk groups, must be mounted before you can add them.

Syntax

ocrconfig -add location_name

Example

ocrconfig -add +data

---

ocrconfig -backuploc

Use the ocrconfig -backuploc command to specify an OCR backup directory location.

Syntax

ocrconfig [-local] -backuploc file_name

Example

ocrconfig –backuploc +bkupdg

---

ocrconfig -delete

Use the ocrconfig -delete command to remove an OCR device or file.

Syntax

ocrconfig -delete file_name

Example

ocrconfig -delete +olddg

---

ocrconfig -downgrade

Use the ocrconfig -downgrade command to downgrade OCR to an earlier specified version.

Syntax

ocrconfig -downgrade [-version version_string]

Example

ocrconfig -downgrade -version 11.1.0.7

---

ocrconfig -export

Use the ocrconfig -export command to export the contents of OCR to a target file.

Syntax

ocrconfig [-local] -export file_name

Example

ocrconfig -export d:\tmp\a

---

ocrconfig -import

Use the ocrconfig -import command to import the contents of a target file into which you exported the contents of OCR back into OCR.

Syntax

ocrconfig [-local] -import file_name

Example

ocrconfig -import d:\tmp\a

---

ocrconfig -manualbackup

Use the ocrconfig -manualbackup command to back up OCR on demand in the location you specify with the -backuploc option.

Syntax

ocrconfig [-local] -manualbackup

Example

ocrconfig -manualbackup

---

ocrconfig -overwrite

Use the ocrconfig -overwrite command to overwrite an OCR configuration in the OCR metadata with the current OCR configuration information that is found on the node from which you run this command.

Syntax

ocrconfig -overwrite

Example

ocrconfig -overwrite

---

ocrconfig -repair

Use the ocrconfig -repair command to repair an OCR configuration on the node from which you run this command. Use this command to add, delete, or replace an OCR location on a node that may have been stopped while you made changes to the OCR configuration in the cluster. OCR locations that you add

must exist, have sufficient permissions, and, in the case of Oracle ASM disk groups, must be mounted before you can add them.

Syntax

ocrconfig -repair -add file_name | -delete file_name | -replace current_file_name -replacement new_file_name

Example

ocrconfig -repair -delete +olddg

---

ocrconfig -replace

Use the ocrconfig -replace command to replace an OCR device or file on the node from which you run this command. OCR locations that you add must exist, have sufficient permissions, and, in the case of Oracle ASM disk groups, must be mounted before you can add them.

Syntax

ocrconfig -replace current_location_name -replacement new_location_name

Example

ocrconfig -replace /oradbocfs/crs/data.ocr -replacement +newdg

---

ocrconfig -restore

Use the ocrconfig -restore command to restore OCR from an automatically created OCR backup file.

Syntax

ocrconfig [-local] -restore file_name

Example

ocrconfig -restore /oradbocfs/crs/BACKUP00.ocr

---

ocrconfig -showbackup

Use the ocrconfig -showbackup command to display the backup location, timestamp, and the originating node name of the backup files. By default, this command displays information for both automatic and manual backups unless you specify auto or manual.

Syntax

ocrconfig [-local] -showbackup [auto | manual]

Example

$ ocrconfig -local -showbackup manual

---

ocrconfig -upgrade

Only root scripts use the ocrconfig -upgrade command to upgrade OCR from a previous version.

---

cluvfy

---

cluvfy comp acfs

Use the cluvfy comp acfs component verification command to check the integrity of Oracle Automatic Storage Management Cluster File System (Oracle ACFS) on all nodes in a cluster.

Syntax

cluvfy comp acfs [-n [node_list] | [all]] [-f file_system] [-verbose]

---

cluvfy comp admprv

Use the cluvfy comp admprv command to verify user accounts and administrative permissions for installing Oracle Clusterware and Oracle RAC software, and for creating an Oracle RAC database or modifying an Oracle RAC database configuration.

Syntax

cluvfy comp admprv [-n node_list] { -o user_equiv [-sshonly] | -o crs_inst [-orainv orainventory_group] |

-o db_inst [-osdba osdba_group] [-fixup] | -o db_config -d oracle_home [-fixup] } [-verbose]

Examples

Example 1: Verifying User Equivalence for All Nodes

cluvfy comp admprv -n all -o user_equiv -verbose

Example 2: Verifying Permissions Required to Install Oracle Clusterware

cluvfy comp admprv -n racnode1,racnode2 -o crs_inst -verbose

Example 3: Verifying Permissions Manage Oracle RAC Databases

cluvfy comp admprv -n all -o db_config -d C:\app\oracle\product\11.2.0\dbhome_1 -fixup -verbose

---

cluvfy comp asm

Use the cluvfy comp asm component verification command to check the integrity of Oracle Automatic Storage Management (Oracle ASM) on all nodes in the cluster. This check ensures that the Oracle ASM instances on the specified nodes are running from the same Oracle home and that asmlib, if it exists, has a valid version and ownership.

Syntax

cluvfy comp asm [-n node_list | all] [-verbose]

Examples

cluvfy comp asm –n all

---

cluvfy comp baseline

Use the cluvfy comp baseline command to capture system and cluster configuration information to create a baseline. You can use this baseline for comparison with the state of the system. You can collect baselines at strategic times, such as after Oracle Clusterware installation, before and after upgrading Oracle Clusterware, or automatically as part of periodic execution of CVU running as an Oracle Clusterware resource. You can also compare several baselines.

Syntax

cluvfy comp baseline -collect {all | cluster | database} [-n node_list] [-db db_unique_name] [-bestpractice | -mandatory] [-binlibfilesonly [-reportname report_name] [-savedir save_dir]]

cluvfy comp baseline -compare baseline1,baseline2,...

Examples

$ cluvfy comp baseline -collect all -n all -db usefdb -bestpractice -report bl1   -savedir /tmp

$ cluvfy comp baseline -compare bl1,bl2

---

cluvfy comp cfs

Use the cluvfy comp cfs component verification command to check the integrity of the clustered file system (OCFS2) you provide using the -f option. CVU checks the sharing of the file system from the nodes in the node list.

Syntax

cluvfy comp cfs [-n node_list] -f file_system [-verbose]

Examples

cluvfy comp cfs -n all -f /u03 -verbose

cluvfy comp cfs -f /oradbshare –n all -verbose

---

cluvfy comp clocksync

Use the cluvfy comp clocksync component verification command to clock synchronization across all the nodes in the node list. CVU verifies a time synchronization service is running (Oracle Cluster Time Synchronization Service (CTSS) or Network Time Protocol (NTP)), that each node is using the same reference server for clock synchronization, and that the time offset for each node is within permissible limits.

Syntax

cluvfy comp clocksync [-noctss] [-n node_list [all]] [-verbose]

---

cluvfy comp clu

Use the cluvfy comp clu component verification command to check the integrity of the cluster on all the nodes in the node list.

Syntax

cluvfy comp clu [-n node_list] [-verbose]

Example

cluvfy comp clu -n all

---

cluvfy comp clumgr

Use the cluvfy comp clumgr component verification command to check the integrity of cluster manager subcomponent, or Oracle Cluster Synchronization Services (CSS), on all the nodes in the node list.

Syntax

cluvfy comp clumgr [-n node_list] [-verbose]

---

cluvfy comp crs

Run the cluvfy comp crs component verification command to check the integrity of the Cluster Ready Services (CRS) daemon on the specified nodes.

Syntax

cluvfy comp crs [-n node_list] [-verbose]

---

cluvfy comp dhcp

Use the cluvfy comp dhcp component verification command to verify that the DHCP server exists on the network and can provide a required number of IP addresses. This verification also verifies the response time for the DHCP server. You must run this command as root.

Syntax

cluvfy comp dhcp -clustername cluster_name [-vipresname application_vip_resource_name]

[-method {sudo|root} [-location directory_path] [-user user_name]] [-networks network_list]

[-port dhcp_port] [-n node_list] [-verbose]

---

cluvfy comp dns

Use the cluvfy comp dns component verification command to verify that the Grid Naming Service (GNS) subdomain delegation has been properly set up in the Domain Name Service (DNS) server.

Syntax

cluvfy comp dns -server -domain gns_sub_domain -vipaddress gns_vip_address] [-port dns_port] [-method {sudo | root}] [-location directory_path] [-user user_name][-verbose] cluvfy comp dns -client -domain gns_sub_domain -vip gns_vip [-method {sudo | root}] [-location directory_path] [-user user_name [-port dns_port] [-last] [-verbose]

---

cluvfy comp freespace

Use the cluvfy comp freespace component verification command to check the free space available in the Oracle Clusterware home storage and ensure that there is at least 5% of the total space available. For example, if the total storage is 10GB, then the check ensures that at least 500MB of it is free.

Syntax

cluvfy comp freespace [-n node_list | all]

---

cluvfy comp gns

Use the cluvfy comp gns component verification command to verify the integrity of the Grid Naming Service (GNS) on the cluster.

Syntax

cluvfy comp gns -precrsinst -domain gns_domain -vip gns_vip [-n node_list] [-verbose]cluvfy comp gns -postcrsinst [-verbose]

---

cluvfy comp gpnp

Use the cluvfy comp gpnp component verification command to check the integrity of Grid Plug and Play on all of the nodes in a cluster.

Syntax

cluvfy comp gpnp [-n node_list] [-verbose]

vahidusefzadeh@gmail.com

---

cluvfy comp ha

Use the cluvfy comp ha component verification command to check the integrity of Oracle Restart on the local node.

Syntax

cluvfy comp ha [-verbose]

---

cluvfy comp healthcheck

Use the cluvfy comp healthcheck component verification command to check your Oracle Clusterware and Oracle Database installations for their compliance with mandatory requirements and best practices guidelines, and to ensure that they are functioning properly.

Syntax

cluvfy comp healthcheck [-collect {cluster|database}] [-db db_unique_name] [-bestpractice|-mandatory] [-deviations] [-html] [-save [-savedir directory_path]]

---

cluvfy comp nodeapp

Use the component cluvfy comp nodeapp command to check for the existence of node applications, namely VIP, NETWORK, and ONS, on all of the specified nodes.

Syntax

cluvfy comp nodeapp [-n node_list] [-verbose]

---

cluvfy comp nodecon

Use the cluvfy comp nodecon component verification command to check the connectivity among the nodes specified in the node list. If you provide an interface list, then CVU checks the connectivity using only the specified interfaces.

Syntax

cluvfy comp nodecon [-n node_list] [-networks network_list] [-verbose]

Example

cluvfy comp nodecon -n node1,node3 –networks eth0 -verbose

---

cluvfy comp nodereach

Use the cluvfy comp nodereach component verification command to check the reachability of specified nodes from a source node.

Syntax

cluvfy comp nodereach -n node_list [-srcnode node] [-verbose]

Example

cluvfy comp nodereach -n node3

---

cluvfy comp ocr

Use the cluvfy comp ocr component verification command to check the integrity of Oracle Cluster Registry (OCR) on all the specified nodes.

Syntax

cluvfy comp ocr [-n node_list] [-verbose]

Example

cluvfy comp ocr

---

cluvfy comp ohasd

Use the cluvfy comp ohasd component verification command to check the integrity of the Oracle High Availability Services daemon.

vahidusefzadeh@gmail.com

Syntax

cluvfy comp ohasd [-n node_list] [-verbose]

Example

cluvfy comp ohasd -n all -verbose

---

cluvfy comp olr

Use the cluvfy comp olr component verification command to check the integrity of Oracle Local Registry (OLR) on the local node.

Syntax

cluvfy comp olr [-verbose]

Example

cluvfy comp olr -verbose

---

cluvfy comp peer

Use the cluvfy comp peer component verification command to check the compatibility and properties of the specified nodes against a reference node. You can check compatibility for non-default user group names and for different releases of the Oracle software. This command compares physical attributes, such as memory and swap space, and user and group values, kernel settings, and installed operating system packages.

Syntax

cluvfy comp peer -n node_list [-refnode node] [-r {10gR1 | 10gR2 | 11gR1 | 11gR2}] [-orainv orainventory_group] [-osdba osdba_group] [-verbose]

Example

cluvfy comp peer -n node1,node2,node4,node7 -verbose

---

cluvfy comp scan

vahidusefzadeh@gmail.com

Use the cluvfy comp scan component verification command to check the Single Client Access Name (SCAN) configuration.

Syntax

cluvfy comp scan [-verbose]

Example

cluvfy comp scan

---

cluvfy comp software

Use the cluvfy comp software component verification command to check the files and attributes installed with the Oracle software.

Syntax

cluvfy comp software [-n node_list] [-d oracle_home] [-r {10.1 | 10.2 | 11.1 | 11.2 | 12.1}] [-verbose]

Example

cluvfy comp software -n all -verbose

---

cluvfy comp space

Use the cluvfy comp space component verification command to check for free disk space at the location you specify in the -l option on all the specified nodes.

Syntax

cluvfy comp space [-n node_list] -l storage_location -z disk_space {B | K | M | G} [-verbose]

Examples

cluvfy comp space -n all -l /home/dbadmin/products –z 5G -verbose

---

cluvfy comp ssa

vahidusefzadeh@gmail.com

Use the cluvfy comp ssa component verification command to discover and check the sharing of the specified storage locations. CVU checks sharing for nodes in the node list.

Syntax

cluvfy comp ssa [-n node_list | -flex -hub hub_list [-leaf leaf_list]][-s storageID_list] [-t {software | data | ocr_vdisk}] [-asm [-asmdev asm_device_list]][-r {10.1 | 10.2 | 11.1 | 11.2 | 12.1}] [-verbose]

Example

cluvfy comp ssa -n all -verbose

---

cluvfy comp sys

Use the cluvfy comp sys component verification command to check that the minimum system requirements are met for the specified product on all the specified nodes.

Syntax

cluvfy comp sys [-n node_list] -p {crs | ha | database} [-r {10gR1 | 10gR2 | 11gR1 | 11gR2}] [-osdba osdba_group] [-orainv orainventory_group] [-fixup] [-verbose]

Examples

cluvfy comp sys -n node1,node2,node3 -p crs -verbose

---

cluvfy comp vdisk

Use the cluvfy comp vdisk component verification command to check the voting files configuration and the udev settings for the voting files on all the specified nodes.

Syntax

cluvfy comp vdisk [-n node_list] [-verbose]

---

cluvfy stage [-pre | -post] acfscfg

Use the cluvfy stage -pre acfscfg command to verify your cluster nodes are set up correctly before configuring Oracle Automatic Storage Management Cluster File System (Oracle ACFS).Use the cluvfy stage -post acfscfg to check an existing cluster after you configure Oracle ACFS.

vahidusefzadeh@gmail.com

Syntax

cluvfy stage -pre acfscfg -n node_list [-asmdev asm_device_list] [-verbose]

cluvfy stage -post acfscfg -n node_list [-verbose]

---

cluvfy stage [-pre | -post] cfs

Use the cluvfy stage -pre cfs stage verification command to verify your cluster nodes are set up correctly before setting up OCFS2.Use the cluvfy stage -post cfs stage verification command to perform the appropriate checks on the specified nodes after setting up OCFS2.

Syntax

cluvfy stage -pre cfs -n node_list -s storageID_list [-verbose]

cluvfy stage -post cfs -n node_list -f file_system [-verbose]

Example

cluvfy stage -pre cfs -n node1,node2,node3,node4 -s /dev/sdd5

---

cluvfy stage [-pre | -post] crsinst

Use the cluvfy stage -pre crsinst command to check the specified nodes before installing Oracle Clusterware. CVU performs additional checks on OCR and voting files if you specify the -c and -q options.

Use the cluvfy stage -post crsinst command to check the specified nodes after installing Oracle Clusterware.

Syntax

cluvfy stage -pre crsinst -n node_list [-r {10.1 | 10.2 | 11.1 | 11.2 | 12.1}] [-c ocr_location_list] [-q voting_disk_list] [-osdba osdba_group] [-orainv orainventory_group] [-asm [-asmgrp asmadmin_group] [-asmdev asm_device_list]] [-crshome Grid_home] [-fixup] [-networks network_list] [-verbose]]

cluvfy stage -post crsinst -n node_list [-verbose]

---

cluvfy stage -pre dbcfg

vahidusefzadeh@gmail.com

Use the cluvfy stage -pre dbcfg command to check the specified nodes before configuring an Oracle RAC database to verify whether your system meets all of the criteria for creating a database or for making a database configuration change.

Syntax

cluvfy stage -pre dbcfg -n node_list -d Oracle_home [-fixup][-servicepwd] [-verbose]

---

cluvfy stage -pre dbinst

Use the cluvfy stage -pre dbinst command to check the specified nodes before installing or creating an Oracle RAC database to verify that your system meets all of the criteria for installing or creating an Oracle RAC database.

Syntax

cluvfy stage -pre dbinst -n node_list [-r {10gR1 | 10gR2 | 11gR1 | 11gR2}][-osdba osdba_group] [-d Oracle_home] [-fixup] [-serviceuser user_name [-servicepwd]][-verbose]

---

cluvfy stage [-pre | -post] hacfg

Use the cluvfy stage -pre hacfg command to check a local node before configuring Oracle Restart.

Use the cluvfy stage -post hacfg command to check the local node after configuring Oracle Restart.

Syntax

cluvfy stage -pre hacfg [-osdba osdba_group] [-orainv orainventory_group][-fixup] [-verbose]

cluvfy stage -post hacfg [-verbose]

---

cluvfy stage -post hwos

Use the cluvfy stage -post hwos stage verification command to perform network and storage verifications on the specified nodes in the cluster before installing Oracle software. This command also checks for supported storage types and checks each one for sharing.

Syntax

cluvfy stage -post hwos -n node_list [-s storageID_list] [-verbose]

---

cluvfy stage [-pre | -post] nodeadd

Use the cluvfy stage -pre nodeadd command to verify the specified nodes are configured correctly before adding them to your existing cluster, and to verify the integrity of the cluster before you add the nodes.

This command verifies that the system configuration, such as the operating system version, software patches, packages, and kernel parameters, for the nodes that you want to add, is compatible with the existing cluster nodes, and that the clusterware is successfully operating on the existing nodes. Run this node on any node of the existing cluster.

Use the cluvfy stage -post nodeadd command to verify that the specified nodes have been successfully added to the cluster at the network, shared storage, and clusterware levels.

Syntax

cluvfy stage -pre nodeadd -n node_list [-vip vip_list] [-fixup] [-verbose]

cluvfy stage -post nodeadd -n node_list [-verbose]

---

cluvfy stage -post nodedel

Use the cluvfy stage -post nodedel command to verify that specific nodes have been successfully deleted from a cluster. Typically, this command verifies that the node-specific interface configuration details have been removed, the nodes are no longer a part of cluster configuration, and proper Oracle ASM cleanup has been performed.

Syntax

cluvfy stage -post nodedel -n node_list [-verbose]

OIFCFG

The Oracle Interface Configuration Tool (OIFCFG) command-line interface helps you to define and administer network interfaces. You can use OIFCFG commands Oracle Clusterware environments to:

- Allocate and deallocate network interfaces to components
- Direct components to use specific network interfaces
- Retrieve component configuration information

vahidusefzadeh@gmail.com

Syntax

oifcfg iflist [-p [-n]]

oifcfg setif {-node nodename | -global} {if_name/subnet:if_type[,if_type]}[,...]

oifcfg getif [-node nodename | -global] [-if if_name[/subnet] [-type if_type]]

oifcfg delif {{-node nodename | -global} [if_name[/subnet]] [-force] | -force}

oifcfg [-help]

Examples

$ oifcfg iflist

$ oifcfg getif

$ oifcfg setif -global eth0/172.19.141.0:cluster_interconnect

$ oifcfg delif -global eth1/172.21.65.0

$ oifcfg delif -global

OLSNODES

The olsnodes command provides the list of nodes and other information for all nodes participating in the cluster. You can use this command to quickly check that your cluster is operational, and all nodes are registered as members of the cluster. This command also provides an easy method for obtaining the node numbers.

Syntax

olsnodes [[-n] [-i] [-s] [-t] [node_name | -l [-p]] | [-c]] [-a] [-g] [-v]

Examples

Example 1: List the VIP addresses for all nodes currently in the cluster

[root@node1]# olsnodes -i

Example 2: List the node names and node numbers for cluster members

[root@node1]# olsnodes -n

Example 3: Display node roles for cluster members

[root@node1]# olsnodes -a

SRVCTL

---

add

The srvctl add command adds the configuration and the Oracle Clusterware applications to OCR for the cluster database, named instances, named services, or for the named nodes. To perform srvctl add operations, you must be logged in as the database administrator and be the Oracle account owner on Linux and UNIX systems, or you must be logged on as a user with Administrator privileges on Windows systems.

When adding an instance, the name that you specify with -instance must match the ORACLE_SID parameter. The database name given with -database db_unique_name must match the DB_UNIQUE_NAME initialization parameter setting. If DB_UNIQUE_NAME is unspecified, then match the DB_NAME initialization parameter setting. The default setting for DB_UNIQUE_NAME uses the setting for DB_NAME. Also, the domain name given with -domain db_domain must match theDB_DOMAIN setting.

---

srvctl add asm

Adds a record for an Oracle ASM instance to the entire cluster. This command must be run only one time from the Oracle Grid Infrastructure home.

Syntax

srvctl add asm [-listener listener_name] [-pwfile password_file_path [-remote [-count {number_of_instances | ALL}] | -proxy]

Example

$ srvctl add asm

---

srvctl add cvu

vahidusefzadeh@gmail.com

Adds the Cluster Verification Utility (CVU) to an Oracle Clusterware configuration.

Syntax

srvctl add cvu [-checkinterval time_in_minutes]

Examples

$ srvctl add cvu -checkinterval 360

---

srvctl add database

Adds a database configuration to Oracle Clusterware.

Syntax

srvctl add database -db db_unique_name [-eval] -oraclehome oracle_home [-node node_name] [-domain domain_name] [-spfile spfile] [-pwfile password_file_path] [-dbtype {RACONENODE | RAC | SINGLE} [-server "server_list"] [-instance instance_name] [-timeout timeout]] [-role {PRIMARY | PHYSICAL_STANDBY | LOGICAL_STANDBY | SNAPSHOT_STANDBY"] [-startoption start_options] [-stopoption stop_options] [-dbname db_name] [-acfspath "acfs_path_list"] [-policy {AUTOMATIC | MANUAL | NORESTART}] [-serverpool "server_pool_list" [-pqpool "pq_pool_list"]] [-diskgroup "disk_group_list"] [-verbose]

Examples

srvctl add database -db crm -oraclehome /u01/oracle/product/12c/mydb -domain example.com -spfile +diskgroup1/crm/spfilecrm.ora -role PHYSICAL_STANDBY -startoption MOUNT -dbtype RAC -dbname crm_psd -policy MANUAL -serverpool "svrpool1,svrpool2" -diskgroup "dgrp1,dgrp2"

---

srvctl add exportfs

Creates an export file system configuration in Oracle Clusterware.

Syntax

srvctl add exportfs -name unique_name -id havip_name -path path_to_export [-clients nfs_client_string] [-options nfs_client_string]

Example

```
# srvctl add exportfs -name export1 -id havip1 -path /u01/db1
```

---

srvctl add filesystem

Adds a device containing a file system (Oracle ACFS or other) to the Oracle Clusterware stack for automount and high availability. This command must be run only one time from the Oracle Grid Infrastructure home.

An Oracle ACFS file system resource is typically created for use with application resource dependency lists. For example, if an Oracle ACFS file system is configured for use as an Oracle Database home, then a resource created for the file system can be included in the resource dependency list of the Oracle Database application. This will cause the file system and stack to be automatically mounted because of the start action of the database application.

To manage Oracle ACFS on Oracle Database 12c installations, use the SRVCTL binary in the Oracle Grid Infrastructure home for a cluster (Grid home). If you have Oracle RAC or Oracle Database installed, then you cannot use the SRVCTL binary in the database home to manage Oracle ACFS.

Syntax

```
srvctl add filesystem -device volume_device -mountpointpath mountpoint_path   [-volume  volume_name] [-diskgroup disk_group_name]    [-node node_list | -serverpool server_pool_list] [-user user_name]   [-fstype {ACFS | EXT3 | EXT4}] [-fsoptions options] [-description description]    [-appid application_id] [-autostart {ALWAYS | NEVER | RESTORE}]
```

Example

```
# srvctl add filesystem -device /dev/asm/d1volume1-295 -mountpath /oracle/cluster1/acfs1
```

To add an Oracle ACFS file system on the dynamic volume device asm-test-55, with this file system mounted on one of the specified nodes at a time:

```
# srvctl add filesystem -fstype ACFS -device asm-test-55 -mountpath myacfs -nodes node1,node2,node3
```

---

srvctl add gns

Use this command to add the Grid Naming Service (GNS) to a cluster when you are using a DHCP public network or to create a client cluster.

Syntax

To add GNS to a cluster that is not currently running GNS:

srvctl add gns [-domain domain_name] -vip {vip_name | ip_address} [-verbose]

To change a cluster that is not running GNS to be a client cluster of another cluster that is running GNS:

srvctl add gns -clientdata file_name

Examples

To add GNS to a cluster, making it a server cluster:

srvctl add gns -vip 192.168.16.17 -domain cluster.mycompany.com

To add GNS to a cluster, making it a client cluster:

srvctl add gns -clientdata /tmp/gnsdata

---

srvctl add havip

Adds highly available VIPs (HAVIPs) (used for highly available NFS exports) to a cluster.

Syntax

srvctl add havip -id havip_name {-address {host_name | ip_address} [-netnum network_number]}

Example

srvctl add havip -id hrexports -address rac_havip -netnum 2

---

srvctl add instance

Adds a configuration for an instance to your cluster database configuration.

You can only use this command for administrator-managed databases. If you have a policy-managed database, then use the srvctl modify srvpoolcommand to add an instance to increase either the maximum size, minimum size, or both, of the server pool used by the database.

Syntax

srvctl add instance -db db_unique_name -instance instance_name -node node_name [-force]

Examples

Examples of this command are:

$ srvctl add instance -db crm -instance crm01 -node gm01

$ srvctl add instance -db crm -instance crm02 -node gm02

$ srvctl add instance -db crm -instance crm03 -node gm03

---

srvctl add listener

Adds a listener to every node in a cluster.

Syntax

To create an Oracle Database listener:

srvctl add listener [-listener listener_name] [-netnum network_number] [-oraclehome Oracle_home] -user

user_name [-endpoints "[TCP:]port_list[/IPC:key] [/NMP:pipe_name][/TCPS:s_port][/SDP:port]"] [-skip]

To create an Oracle ASM listener:

srvctl add listener [-listener listener_name] -asmlistener [-subnet subnet]

[-endpoints "[TCP:]port_list[/IPC:key][/NMP:pipe_name][/TCPS:s_port][/SDP:port]"] [-skip]

To create a Leaf listener:

srvctl add listener [-listener listener_name] -leaflistener [-subnet subnet] [-endpoints "[TCP:]port_list[/IPC:key][/NMP:pipe_name][/TCPS:s_port][/SDP:port]"] [-skip]

To create a SCAN listener, use the srvctl add scan_listener command.

Example

The following command adds a listener named listener112 that is listening on ports 1341, 1342, and 1345 and runs from the Oracle home directory on every node in the cluster:

```
$ srvctl add listener -listener listener112 -endpoints "1341,1342,1345" -oraclehome
/u01/app/oracle/product/12.1.0/db1
```

---

srvctl add mgmtdb

Adds a management database (CHM repository) resource to the cluster.

Syntax

```
srvctl add mgmtdb [-domain domain]
```

Example

```
$ srvctl add mgmtdb -domain example.com
```

---

srvctl add mgmtlsnr

Adds a management listener resource (for CHM) to the cluster.

Syntax

```
srvctl add mgmtlsnr [-endpoints "[TCP:]port_list[/IPC:key][/NMP:pipe_name] [/TCPS:s_port][/SDP:port]"] [-skip]
```

Example

The following command adds a management listener that is listening on port 1341 to the cluster:

```
$ srvctl add mgmtlsnr -endpoints "TCP:1341"
```

---

srvctl add network

Adds a static or dynamic network. If your server connects to more than one network, then you can use this command to configure an additional network interface for Oracle RAC, allowing you to create VIPs

on multiple public networks. You can also use the LISTENER_NETWORKS database initialization parameter to control client redirects to the appropriate network

Syntax

srvctl add network [-netnum net_number] -subnet subnet/netmask[/if1[|if2|...]] [-nettype {static | dhcp | autoconfig | mixed}] [-leaf] [-verbose]

Example

srvctl add network -netnum 3 -subnet 192.168.3.0/255.255.255.0

---

srvctl add nodeapps

Adds a node application configuration to the specified node.

Syntax

srvctl add nodeapps {-node node_name -address {vip_name | ip_address}/netmask[/if1[|if2|..]] [-skip]} [-emport em_port] [-onslocalport ons_local_port] [-onsremoteport ons_remote_port] [-onshostport hostname_port_list] [-remoteservers hostname_port_list [-verbose]

Example

srvctl add nodeapps -node crmnode1 -address 1.2.3.4/255.255.255.0

---

srvctl add oc4j

Adds an OC4J instance to all the nodes in the cluster.

Syntax

srvctl add oc4j [-verbose]

Example

srvctl add oc4j

---

vahidusefzadeh@gmail.com

srvctl add ons

Adds an Oracle Notification Service daemon to an Oracle Restart configuration.

Note:

This command is only available with Oracle Restart.

Syntax

srvctl add ons [-l ons_local_port] [-r ons_remote_port] [-t host[:port][,host[:port]][...]] [-v]

Example

$ srvctl add ons -l 6200

---

srvctl add scan

Adds Oracle Clusterware resources for the given SCAN. This command creates the same number of SCAN VIP resources as the number of IP addresses that SCAN resolves to, or 3 when network_number identifies a dynamic network and Oracle GNS configuration. For static networks, the addresses to which the SCAN resolves in DNS must match the address type of the subnet. For an IPv4 network, the SCAN must resolve to IPv4 addresses.

Syntax

srvctl add scan -scanname scan_name [-netnum network_number

Example

srvctl add scan -scanname scan.mycluster.example.com

---

srvctl add scan_listener

Adds Oracle Clusterware resources for the SCAN listeners. The number of SCAN listener resources created is the number of SCAN VIP resources.

Syntax

srvctl add scan_listener [-listener lsnr_name_prefix] [-skip] [-enpoints "[TCP:]port_list[/IPC:key][/NMP:pipe_name][/TCPS:s_port][/SDP:port]"] [-invitednodes node_list] [-invitedsubnets subnet_list]

Example

srvctl add scan_listener -listener myscanlistener

---

srvctl add service

Adds services to a database and assigns them to instances. If you have multiple instances of a cluster database on the same node, then always use only one instance on that node for all of the services that node manages.

Note:

The srvctl add service command does not accept placement parameters for Oracle RAC One Node databases.

Syntax

srvctl add service -db db_unique_name -service service_name [-eval]

-serverpool server_pool [-cardinality {UNIFORM | SINGLETON}] [-edition edition_name] [-netnum network_number] [-role "[PRIMARY][,PHYSICAL_STANDBY] [,LOGICAL_STANDBY] [,SNAPSHOT_STANDBY]" [-policy {AUTOMATIC | MANUAL}] [-notification {TRUE | FALSE}] [-clbgoal {SHORT | LONG}] [-failovertype {NONE|SESSION|SELECT|TRANSACTION}] [-rlbgoal {NONE | SERVICE_TIME | THROUGHPUT}] [-dtp {TRUE | FALSE}] [-failovermethod {NONE | BASIC}] [-failoverretry failover_retries] [-failoverdelay failover_delay] [-pdb pluggable_database] [-sql_translation_profile sql_translation_profile] [-global {TRUE | FALSE}] [-maxlag max_lag_time] [-commit_outcome {TRUE|FALSE}] [-retention retention_time] [-replay_init_time replay_initiation_time] [-session_state {STATIC | DYNAMIC}] [-pqservice pq_service] [-pqpool pq_pool_list] [-force]

To add a service to an administrator-managed database:

srvctl add service -database db_unique_name -service service_name [-eval] -preferred preferred_list -available available_list] [-netnum network_number] [-tafpolicy {BASIC | NONE | PRECONNECT}] [-edition edition_name] [-role "[PRIMARY] [,PHYSICAL_STANDBY][,LOGICAL_STANDBY] [,SNAPSHOT_STANDBY]" [-policy {AUTOMATIC | MANUAL}] [-notification {TRUE | FALSE}] [-clbgoal {SHORT | LONG}] [-failovertype {NONE|SESSION|SELECT|TRANSACTION}] [-rlbgoal {NONE | SERVICE_TIME | THROUGHPUT}] [-dtp {TRUE | FALSE}] [-failovermethod {NONE | BASIC}] [-failoverretry failover_retries] [-failoverdelay failover_delay] [-pdb pluggable_database] [-sql_translation_profile sql_translation_profile] [-global {TRUE | FALSE}] [-maxlag max_lag_time] [-commit_outcome {TRUE|FALSE}] [-retention retention_time] [-replay_init_time replay_initiation_time] [-session_state {STATIC|DYNAMIC}] [-pqservice pq_service] [-pqpool pq_pool_list] [-force] [-verbose]

To update the preferred and available lists of an existing service:

srvctl add service -db db_unique_name -service service_name -newinst {-prefered preferred_list | -available available_list} [-force] [-verbose]

Examples

Use this example syntax to add the gl.example.com service to the my_rac database with Fast Application Notification enabled for OCI connections, a failover method of BASIC, a Connection Load Balancing Goal of LONG, a failover type of SELECT, and 180 failover retries with a failover delay of 5 seconds:

srvctl add service -db my_rac -service gl.example.com -notification TRUE \ -failovermethod BASIC -failovertype SELECT -failoverretry 180 -failoverdelay 5 \ -clbgoal LONG

Use this example syntax to add a named service to a database with preferred instances and available instances and enabled for TAF:

srvctl add service -db crm -service sales -preferred crm01,crm02 -available crm03\ -tafpolicy BASIC

---

srvctl add srvpool

Adds a server pool that is configured to host Oracle databases to a cluster.

vahidusefzadeh@gmail.com

Syntax

srvctl add srvpool -serverpool server_pool_name [-eval] [-importance importance] [-min min_size] [-max max_size] [-servers "node_list" | -category server_category] [-force] [-verbose]

Example

$ srvctl add srvpool -serverpool SP1 -importance 1 -min 3 -max 7

---

srvctl add vip

Adds a VIP to a node.

Syntax

srvctl add vip -node node_name -address {VIP_name|ip}/netmask[/if1[|if2|...]] -netnum network_number [-skip] [-verbose]

Example

srvctl add network -netnum 2 -subnet 192.168.16.0/255.255.255.0

srvctl add vip -node node7 -address 192.168.16.17/255.255.255.0 -netnum 2

---

config

The srvctl config command displays the configuration stored in the Oracle Clusterware resource attributes.

Note:

If you disabled an object for which you are trying to obtain configuration information using the srvctl disable object -nnode_name command, then remember that using the srvctl disable object -n node_name command on a per-node basis in your cluster to disable an object throughout the cluster is different from globally disabling an object using the srvctl disableobject command without the -n parameter. In the former case, the srvctl config object command may report that the object is still enabled.

---

vahidusefzadeh@gmail.com

srvctl config asm

Displays the configuration for all Oracle ASM instances.

Syntax

srvctl config asm [-proxy] [-detail]

Example

$ srvctl config asm

---

srvctl config cvu

Displays the time interval, in minutes, between CVU checks.

---

srvctl config database

Displays the configuration for an Oracle RAC database or lists all configured databases that are registered with Oracle Clusterware.

Syntax

srvctl config database [-db db_unique_name] [-all] [-verbose]

Example

$ srvctl config database -db myDB

---

srvctl config exportfs

Displays the configuration for an export file system in Oracle Clusterware.

Syntax

srvctl config exportfs [-name unique_name | -id havip_name]

Example

$ srvctl config exportfs -id havip1

---

srvctl config filesystem

Displays the configuration for a specific file system resource.

Syntax

srvctl config filesystem -device volume_device

Examples

To list the configuration of all file systems, use the following example:

srvctl config filesystem

To show the configuration for a specific device, use the following example:

srvctl config filesystem -device /dev/asm/d1volume1-295

---

srvctl config gns

Displays the configuration for GNS.

Note:

This command can be used only with Oracle Clusterware.

Syntax

srvctl config gns [-subdomain] [-multicastport] [-node node_name] [-port] [-status] [-version] [-query name] [-list] [-clusterguid] [-clustername] [-clustertype] [-loglevel] [-network] [-detail]

---

srvctl config havip

Displays configuration information for a specific highly available VIP (HAVIP) (used for highly available NFS exports) or all HAVIP resources in a cluster.

Syntax

srvctl config havip [-id havip_name]

Example

$ srvctl config havip -id hrexports

VIP: /mjk-vm3-vip/10.149.237.196/10.149.236.0/255.255.252.0/eth0

Description: HR Exports

---

srvctl config listener

Displays configuration information of a specific listener that is registered with Oracle Clusterware.

Syntax

srvctl config listener [-listener listener_name | -asmlistener | -leaflistener] [-all]

---

srvctl config mgmtdb

Displays configuration information for the management database (CHM repository) resource.

Syntax

srvctl config mgmtdb [-verbose] [-all]

---

srvctl config mgmtlsnr

Displays configuration information for the management listener resource (for CHM).

Syntax

srvctl config mgmtlsnr [-all]

---

srvctl config network

Displays the network configuration for the cluster.

vahidusefzadeh@gmail.com

Syntax

srvctl config network [-netnum network_number]

Example

$ srvctl config network -netnum 2

---

srvctl config nodeapps

Displays the VIP configuration for each node in the cluster.

Syntax

srvctl config nodeapps [-viponly] [-gsdonly] [-onsonly]

Example

$ srvctl config nodeapps -viponly -gsdonly -ons

---

srvctl config oc4j

Displays configuration information for the OC4J instance.

---

srvctl config oc4j

This command has no parameters.

---

srvctl config ons

Displays configuration information for the Oracle Notification Service daemon.

Note:

This command is only available with Oracle Restart.

---

vahidusefzadeh@gmail.com

srvctl config ons

This command has no parameters.

---

srvctl config scan

Displays the configuration information for all SCAN VIPs, by default, or a specific SCAN VIP identified by ordinal_number.

Syntax

srvctl config scan [-scannumber ordinal_number]

Example

$ srvctl config scan -scannumber 1

---

srvctl config scan_listener

Displays the configuration information for all SCAN listeners, by default, or a specific listener identified by ordinal_number.

Syntax

srvctl config scan_listener [-scannumber ordinal_number]

Example

$ srvctl config scan_listener -scannumber 1

---

srvctl config service

Displays the configuration for a service.

Syntax

srvctl config service -db db_unique_name [-service service_name] [-verbose]

Examples

$ srvctl config service -db crm -service webapps

This command displays information similar to the following for a administrator-managed database:

$ srvctl config service -db crm -service webapps

---

srvctl config srvpool

Displays configuration information including name, minimum size, maximum size, importance, and a list of server names, if applicable, for a specific server pool in a cluster.

Syntax

srvctl config srvpool [-serverpool pool_name]

Example

$ srvctl config srvpool -serverpool dbpool

---

srvctl config vip

Displays all VIPs on all networks in the cluster except for user VIPs.

Syntax

srvctl config vip {-node node_name | -vip vip_name}

Example

$ srvctl config vip -node crmnode1

VIP exists: /crmnode1-vip/192.168.2.20/255.255.255.0/eth0

---

srvctl config volume

Displays the configuration for a specific volume or all volumes.

Syntax

srvctl config volume [-volume volume_name] [-diskgroup disk_group_name] [-device volume_device]

vahidusefzadeh@gmail.com

Examples

$ srvctl config volume -device /dev/asm/volume1-123

If you do not specify any parameters, then SRVCTL displays configuration information for all volumes, similar to the following:

$ srvctl config volume

---

srvctl convert database

The srvctl convert database command converts a database either to or from an Oracle RAC One Node database.

Syntax

srvctl convert database -db db_unique_name -dbtype RACONENODE [-instance instance_name] [-timeout timeout]

srvctl convert database -db db_unique_name -dbtype RAC [-node node_name]

Example

$ srvctl convert database -db myDB -dbtype RACONENODE -instance myDB3

---

disable

Disables a specified object (cluster database, database instance, Oracle ASM instance, or service). Use the srvctl disable command when you must shut down an object for maintenance. The disabled object does not automatically restart.

When you issue the disable command, the object is disabled and unavailable to run under Oracle Clusterware for automatic startup, failover, or restart. Additionally, you cannot run the srvctl start command on a disabled object until you first re-enable the object. If you specify -instance instance_name or -node node_name, then SRVCTL only disables the object on the specified instance or node.

srvctl disable asm

Disables the Oracle ASM proxy resource. Oracle ASM will restart if it contains Oracle Clusterware data or if the node restarts and it was running before the node failed. The srvctl disable asm command also prevents the srvctl start asm command from starting Oracle ASM instances.

Disabling the Oracle ASM proxy resource prevents the databases and disk groups from starting because they depend on the Oracle ASM proxy resource.

Syntax

srvctl disable asm [-proxy] [-node node_name]

Example

$ srvctl disable asm -node crmnode1

---

srvctl disable cvu

Disable the Cluster Verification Utility (CVU) for Oracle Clusterware management, if enabled.

Syntax

srvctl disable cvu [-node node_name]

Examples

$ srvctl disable cvu -node crmnode1

---

srvctl disable database

Disables a database. If the database is a cluster database, then its instances are also disabled.

Syntax

srvctl disable database -db db_unique_name [-node node_name]

Example

$ srvctl disable database -db mydb1

---

vahidusefzadeh@gmail.com

srvctl disable diskgroup

Disables a specific disk group on a number of specified nodes.

Syntax

srvctl disable diskgroup -diskgroup diskgroup_name [-node "node_list"]

Example

$ srvctl disable diskgroup -diskgroup dgroup1 -node "mynode1,mynode2"

---

srvctl disable exportfs

Disables an export file system managed by Oracle Clusterware.

Syntax

srvctl disable exportfs -name filesystem_name

Example

srvctl disable exportfs -name export1

---

srvctl disable filesystem

Disables a specific Oracle Clusterware-managed file system volume.

Syntax

srvctl disable filesystem -device volume_device

Example

srvctl disable filesystem -device /dev/asm/racvol1

---

srvctl disable gns

Disables GNS for a specific node, or all available nodes in the cluster.

Syntax

srvctl disable gns [-node node_name] [-verbose]

Example

$ srvctl disable gns -node crm7

---

srvctl disable havip

Prevents a specific highly available VIP (HAVIP) (used for highly available NFS exports) resource from running on a number of specified nodes.

Syntax

srvctl disable havip -id havip_name [-node node1,node2,...,noden]

Example

$ srvctl disable havip -id hrexports -node myNode1,myNode3

---

srvctl disable instance

Disables an instance. If the instance that you disable with this command is the last enabled instance, then this operation also disables the database.

Syntax

srvctl disable instance -db db_unique_name -instance instance_name_list

Example

$ srvctl disable instance -db crm -instance "crm1,crm3"

---

srvctl disable listener

Disables a listener resource.

Syntax

srvctl disable listener [-listener listener_name] [-node node_name]

Example

$ srvctl disable listener -listener listener_crm -node node5

---

srvctl disable mgmtdb

Disables the management database (CHM repository) resource on the specified node.

Syntax

srvctl disable mgmtdb [-node node_name]

Example

$ srvctl disable mgmtdb

---

srvctl disable mgmtlsnr

Disables the management listener resource (for CHM) on the specified node.

Syntax

srvctl disable mgmtlsnr [-node node_name]

Example

$ srvctl disable mgmtlsnr

---

srvctl disable nodeapps

Disables node applications on all nodes in the cluster.

Syntax

srvctl disable nodeapps [-gsdonly] [-adminhelper] [-verbose]

Example

$ srvctl disable nodeapps -gsdonly -verbose

---

srvctl disable oc4j

Disables the OC4J instance on all nodes or on a specific node.

Syntax

srvctl disable oc4j [-node node_name] [-verbose]

Example

$ srvctl disable oc4j -node crm3

---

srvctl disable ons

Disables the Oracle Notification Service daemon for Oracle Restart installations.

srvctl disable ons [-verbose]

The only parameter for this command is -verbose, which indicates that verbose output should be displayed.

---

srvctl disable scan

Disables all SCAN VIPs, by default, or a specific SCAN VIP identified by ordinal_number.

Syntax

srvctl disable scan [-scannumber ordinal_number]

Example

$ srvctl disable scan -scannumber 1

---

srvctl disable scan_listener

Disables all SCAN listeners, by default, or a specific listener identified by ordinal_number.

vahidusefzadeh@gmail.com

Syntax

srvctl disable scan_listener [-scannumber ordinal_number]

Example

$ srvctl disable scan_listener -scannumber 1

---

srvctl disable service

Disables a service. Disabling an entire service affects all of the instances, disabling each one. When the entire service is already disabled, a srvctl disable service operation on the entire service affects all of the instances and disables them; it just returns an error. This means that you cannot always use the entire set of service operations to manipulate the service indicators for each instance.

Syntax

srvctl disable service -db db_unique_name -servics "service_name_list" [-instance instance_name | -node node_name]

Example

$ srvctl disable service -db crm -service "crm,marketing"

The following example disables a service for the CRM database that is running on the CRM1 instance, resulting in the service still being available for the database, but on one less instance:

$ srvctl disable service -db crm -service crm -instance crm1

---

srvctl disable vip

Disables a specific VIP.

Syntax

srvctl disable vip -vip vip_name [-verbose]

Example

$ srvctl disable vip -vip vip1 -verbose

---

vahidusefzadeh@gmail.com

srvctl disable volume

Disables Oracle Clusterware management for a specific volume or all volumes.

This command allows a volume device to be stopped by operating on the Oracle Clusterware resource for the volume. This command does not stop volume device.

Syntax

srvctl disable volume {-volume volume_name -diskgroup disk_group_name | -device volume_device}

Example

$ srvctl disable volume -volume VOLUME1 -diskgroup DATA

---

srvctl downgrade database

The srvctl downgrade database command downgrades the configuration of a database and its services from its current version to the specified lower version.

Syntax

srvctl downgrade database -db db_unique_name -oraclehome Oracle_home -targetversion to_version

---

---

srvctl enable asm

Enables an Oracle ASM instance.

Syntax

srvctl enable asm [-proxy] [-node node_name]

Example

$ srvctl enable asm -node crmnode1

---

vahidusefzadeh@gmail.com

srvctl enable cvu

Enable the Cluster Verification Utility (CVU) for Oracle Clusterware management if disabled.

Syntax

srvctl enable cvu [-node node_name]

Examples

$ srvctl enable cvu -node crmnode1

---

srvctl enable database

Enables a cluster database and its instances.

Syntax

srvctl enable database -db db_unique_name [-node node_name]

Example

$ srvctl enable database -db mydb1

---

srvctl enable diskgroup

Enables a specific disk group on a number of specified nodes.

Syntax

srvctl enable diskgroup -diskgroup diskgroup_name [-node "node_list"]

Example

$ srvctl enable diskgroup -diskgroup diskgroup1 -node "mynode1,mynode2"

---

srvctl enable exportfs

Enables an export file system configuration in Oracle Clusterware.

vahidusefzadeh@gmail.com

Syntax

srvctl enable exportfs -name filesystem_name

Example

srvctl enable exportfs -name export1

---

srvctl enable filesystem

Enables an Oracle ACFS volume or generic file system in Oracle Clusterware.

Syntax

srvctl enable filesystem -device volume_device

Example

srvctl enable filesystem -device /dev/asm/racvol1

---

srvctl enable gns

Enables GNS on all nodes or a specific node.

Syntax

srvctl enable gns [-node node_name] [-verbose]

Example

$ srvctl enable gns

---

srvctl enable havip

Enables a specific highly available VIP (HAVIP) (used for highly available NFS exports) to run on a number of specified nodes.

Syntax

srvctl enable havip -id havip_name [-node node_name]

Example

srvctl enable havip -id hrexports -node myNode1

---

srvctl enable instance

Enables an instance for an Oracle RAC database. If you use this command to enable all instances, then the database is also enabled.

Syntax

srvctl enable instance -db db_unique_name -instance "instance_name_list"

Example

$ srvctl enable instance -db crm -instance "crm1,crm2"

---

srvctl enable listener

Enables a listener resource.

Syntax

srvctl enable listener [-listener listener_name] [-node node_name]

Example

$ srvctl enable listener -listener listener_crm -node node5

---

srvctl enable mgmtdb

Enables a management database (CHM repository) resource.

Syntax

srvctl enable mgmtdb [-node node_name]

Example

$ srvctl enable mgmtdb -node node5

---

srvctl enable mgmtlsnr

Enables a management database listener resource (for CHM).

Syntax

srvctl enable mgmtlsnr [-node node_name]

Example

$ srvctl enable mgmtlsnr -node node5

---

srvctl enable nodeapps

Enables the node applications on all nodes in the cluster.

Syntax

srvctl enable nodeapps [-gsdonly] [-adminhelper] [-verbose]

Example

$ srvctl enable nodeapps -gsdonly -verbose

---

srvctl enable oc4j

Enables the OC4J instance on all nodes or on a specific node.

Syntax

srvctl enable oc4j [-node node_name] [-verbose]

Example

$ srvctl enable oc4j -node crm3

vahidusefzadeh@gmail.com

---

srvctl enable ons

Enables the Oracle Notification Service daemon.

Note:

This command is only available with Oracle Restart.

Syntax

srvctl enable ons [-verbose]

Example

$ srvctl enable ons

---

srvctl enable scan

Enables all SCAN VIPs, by default, or a specific SCAN VIP identified by its ordinal_number.

Syntax

srvctl enable scan [-scannumber ordinal_number]

Example

$ srvctl enable scan -scannumber 1

---

srvctl enable scan_listener

Enables all SCAN listeners, by default, or a specific listener identified by its ordinal_number.

Syntax

srvctl enable scan_listener [-scannumber ordinal_number]

Example

$ srvctl enable scan_listener -scannumber 1

vahidusefzadeh@gmail.com

---

srvctl enable service

Enables a service for Oracle Clusterware. Enabling an entire service also affects the enabling of the service over all of the instances by enabling the service at each one. When the entire service is already enabled, an srvctl enable service operation does not affect all of the instances and enable them. Instead, this operation returns an error. Therefore, you cannot always use the entire set of service operations to manipulate the service indicators for each instance.

Syntax

srvctl enable service -db db_unique_name -service "service_name_list" [-instance instance_name | -node node_name]

Examples

The following example globally enables a service:

$ srvctl enable service -db crm -service crm

The following example enables a service to use a preferred instance:

$srvctl enable service -db crm -service crm -instance crm1

---

srvctl enable vip

Enables a specific VIP.

Syntax

srvctl enable vip -vip vip_name [-verbose]

Example

$ srvctl enable vip -vip crm1-vip -verbose

---

srvctl enable volume

Enables Oracle Clusterware management for a specific volume or all volumes.

This command allows a volume device to be started by operating on the Oracle Clusterware resource for the volume. This command does not start the volume device, and is different from the SQL command ALTER DISKGROUP ENABLE VOLUME or the ASMCMD command volenable, because these two commands bring the volume device online, in a running state, making the volume device accessible.

Syntax

srvctl enable volume {-volume volume_name -diskgroup disk_group_name | -device volume_device}

Example

$ srvctl enable volume -volume VOLUME1 -diskgroup DATA

---

export

The srvctl export gns command exports Grid Naming Service (GNS) instance data to a file that you can use when you are either moving GNS from one server cluster to another or when you are creating a client cluster

---

srvctl export gns

The srvctl export gns command exports Grid Naming Service (GNS) instance data to a file that you can use when you are either moving GNS from one server cluster to another or when you are creating a client cluster.

SRVCTL extracts the data from OCR. Exported data includes:

The credentials used to authorize users, which includes the VIP address on which the server listens

The names and DNS records kept in GNS

Syntax

srvctl export gns {-instance path_to_file | -clientdata path_to_file}

Example

srvctl export gns -clientdata /temp/gnsdata/gns.txt

---

vahidusefzadeh@gmail.com

srvctl getenv asm

Displays the values for environment variables associated with Oracle ASM.

Syntax

srvctl getenv asm [-envs "name_list"]

Example

$ srvctl getenv asm

---

srvctl getenv database

Displays the values for environment variables associated with a database.

Syntax

srvctl getenv database -db db_unique_name [-envs "name_list"]

Example

$ srvctl getenv database -db crm

---

srvctl getenv listener

Gets the environment variables for the specified listener.

Syntax

srvctl getenv listener [-listener listener_name] [-envs "name_list"]

Example

$ srvctl getenv listener

---

srvctl getenv mgmtdb

Gets the environment variables for the management database (CHM repository).

vahidusefzadeh@gmail.com

Syntax

srvctl getenv mgmtdb [-envs "name_list"]

Example

$ srvctl getenv mgmtdb

---

srvctl getenv mgmtlsnr

Gets the environment variables for the management listener resource (for CHM).

Syntax

srvctl getenv mgmtlsnr [-envs "name_list"]

Example

$ srvctl getenv mgmtlsnr

---

srvctl getenv nodeapps

Gets the environment variables for the node application configurations.

Syntax

srvctl getenv nodeapps [-viponly] [-gsdonly] [-onsonly] [-envs "name_list"] [-verbose]

Example

$ srvctl getenv nodeapps -viponly

---

srvctl getenv vip

Gets the environment variables for the specified VIP.

Syntax

srvctl getenv vip -vip vip_name [-envs "name_list"] [-verbose]

Example

$ srvctl getenv vip -vip node1-vip

---

import

The srvctl import command imports data from a file that you create when you run the srvctl export gns command. Use this command when you want to locate GNS to a different server cluster.

---

srvctl import gns

The srvctl import command imports data from a file that you create when you run the srvctl export gns command. Use this command when you want to locate GNS to a different server cluster.

When you import GNS data, SRVCTL stores the credentials and places the record data into OCR. If another GNS instance is running or data for another instance is encountered during the import procedure, then an error occurs.

This command also makes the cluster in which you run it the server cluster.

Syntax

srvctl import gns -instance path_to_file

Example

srvctl import gns -instance /temp/gnsdata/gns.txt

---

modify

Enables you to modify the instance configuration without removing and adding Oracle Clusterware resources. Using modify preserves the environment in the OCR configuration that would otherwise need to be reentered. The configuration description is modified in the OCR configuration, and a new Oracle Clusterware profile is generated and registered. The change takes effect when the application is next restarted.

---

srvctl modify asm

Modify the listener used by Oracle ASM, the disk group discovery string used by Oracle ASM, or the SPFILE used by Oracle ASM for a noncluster database or a cluster database.

Syntax

srvctl modify asm [-listener listener_name] [-pwfile password_file_path] [-diskstring asm_diskstring] [-spfile spfile_path_name [-proxy]]

Example

$ srvctl modify asm -listener lsnr1

---

srvctl modify cvu

Modifies the check interval for CVU in an Oracle Clusterware configuration.

Syntax

srvctl modify cvu [-checkinterval time_in_minutes]]

Example

$ srvctl modify cvu -checkinterval 240

---

srvctl modify database

Modifies the configuration for a database.

Syntax

srvctl modify database -db db_unique_name [-eval] [-dbname db_name] [-instance instance_name] [-oraclehome oracle_home] [-user user_name] [-server server_list] [-timeout timeout] [-domain db_domain] [-spfile spfile] [-pwfile password_file_path] [-role {PRIMARY|PHYSICAL_STANDBY|LOGICAL_STANDBY|SNAPSHOT_STANDBY}] [-startoption start_options] [-stopoption stop_options] [-policy {AUTOMATIC | MANUAL | NORESTART}] [-

serverpool "server_pool_list" [-node node_name]] [-pqpool pq_server_pools] [{-diskgroup "diskgroup_list" | -nodiskgroup}] [-acfspath "acfs_path_list"] [-force]

Examples

The following example changes the role of a database to a logical standby:

$ srvctl modify database -db crm -role logical_standby

The following example directs the racTest database to use the SYSFILES, LOGS, and OLTP Oracle ASM disk groups.

$ srvctl modify database -db racTest -diskgroup "SYSFILES,LOGS,OLTP"

---

srvctl modify exportfs

Modifies an export file system configuration in Oracle Clusterware.

Syntax

srvctl modify exportfs -name unique_name [-path path_to_export][-clients node_list] [-options nfs_options_string]

Example

srvctl modify exportfs -name exportfs1 -path /mnt/racdb1

---

srvctl modify filesystem

Modifies the configuration of the file system resource.

Syntax

srvctl modify filesystem -device volume_device -user user_name [-path mount_point_path] [-node node_list | -serverpool server_pool_list] [-fsoptions options] [-description description] [-autostart {ALWAYS | NEVER | RESTORE}]

Example

srvctl modify filesystem -device /dev/asm/racvol1 -user sysad

---

srvctl modify gns

Modifies the IP address, domain, or other configuration parameters used by GNS.

Syntax

srvctl modify gns -loglevel log_level srvctl modify gns [-resolve address] [-verify name] [-parameter parameter:value[,parameter:value...]] [-vip vip_address] [-clientdata file_name] [-forward domain_list] [-refused domain_list] [-excluded interface_list] [-verbose]

Example

$ srvctl modify gns -vip 192.0.2.15

---

srvctl modify havip

Modifies a highly available VIP (HAVIP) (used for highly available NFS exports).

Syntax

srvctl modify havip -id havip_name [-address {host_name | ip_address} [-netnum network_number]] [-description text]

Example

srvctl modify havip -id hrexports -address 192.168.16.17 -netnum 2

---

srvctl modify instance

For an administrator-managed database, this command modifies the configuration for a database instance from its current node to another node. For a policy-managed database, this command defines an instance name to use when the database runs on the specified node.

Syntax

srvctl modify instance -db db_unique_name -instance instance_name -node node_name

Examples

The following example to changes the configuration of an administrator-managed database, amdb, so that the database instance, amdb1, runs on the specified node, mynode:

```
$ srvctl modify instance -db amdb -instance amdb1 -node mynode
```

The following example causes the policy-managed database pmdb, when and if it runs on mynode, to use the instance name pmdb1:

```
$ srvctl modify instance -db pmdb -instance pmdb1 -node mynode
```

The following example removes the directive established by the previous example:

```
$ srvctl modify instance -db pmdb -instance pmdb1 -node ""
```

---

srvctl modify listener

Changes the Oracle home directory from which the listener runs, the name of the operating system user who owns Oracle home directory from which the listener runs, the listener endpoints, or the public subnet on which the listener listens, either for the default listener, or a specific listener, that is registered with Oracle Restart or with Oracle Clusterware.

If you want to change the name of a listener, then use the srvctl remove listener and srvctl add listener commands.

Syntax

```
srvctl modify listener [-listener listener_name] [-oraclehome oracle_home]  [-user user_name] [-netnum network_number] [-endpoints "[TCP:]port_list[/IPC:key][/NMP:pipe_name][/TCPS:s_port][/SDP:port]"]
```

Example

```
$ srvctl modify listener -endpoints "TCP:1521,1522"
```

---

srvctl modify mgmtdb

Modifies the configuration for the management database (CHM repository).

Syntax

srvctl modify mgmtdb [-pwfile password_file_path] [-spfile spfile_path] [-startoption start_option] [-stopoption stop_option] [-diskgroup "diskgroup_list"]

Example

$ srvctl modify mgmtdb -diskgroup "SYSFILES"

---

srvctl modify mgmtlsnr

Modifies the configuration for the management listener resource (for CHM).

Syntax

srvctl modify mgmtlsnr [-endpoints "[TCP:]port_list[/IPC:key][/NMP:pipe_name] [/TCPS:s_port][/SDP:port]"]

Example

$ srvctl modify mgmtlsnr -endpoints "TCP:2521,2522"

---

srvctl modify network

Modifies the subnet, network type, or IP address type for a network.

Syntax

srvctl modify network [-netnum network_number] [-subnet subnet/netmask [/if1[|if2|...]]] [-nettype network_type | -iptype {ipv4 | ipv6 | both]} [-verbose]

Examples

The following example changes the subnet number, netmask, and interface list:

srvctl modify network -subnet 192.168.2.0/255.255.255.0/eth0

The following example changes the second network to DHCP:

srvctl modify network -netnum 2 -nettype dhcp

The following example adds an IPv6 subnet and netmask to the default network:

srvctl modify network -subnet 2606:b400:400:18c0::/64

The following example removes the IPv4 configuration from a network:

srvctl modify network -iptype ipv6

---

srvctl modify nodeapps

Modifies the configuration for a node application.

Syntax

srvctl modify nodeapps {[-node node_name -address {vip_name|vip_address}/ netmask[/if1[|if2|...]] [-skip]] [-nettype network_type] [-emport em_port] [-onslocalport ons_local_port] [-onsremoteport ons_remote_port] [-remoteservers host:[port][,host:port,...]] [-verbose]

srvctl modify nodeapps [-subnet subnet/netmask[/if1[|if2|...]] [-nettype network_type] [-emport em_port] [-onslocalport ons_local_port] [-onsremoteport ons_remote_port] [-remoteservers host:[port][,host:port,...]] [-verbose]

Example

The following example changes the nodeapps resource on mynode1 to use the application VIP of 100.200.300.40 with a subnet mask of 255.255.255.0 on the network interface eth0:

$ srvctl modify nodeapps -node mynode1 -addr 100.200.300.40/255.255.255.0/eth0

---

srvctl modify oc4j

Modifies the RMI port for the OC4J instances.

Syntax

srvctl modify oc4j -rmiport port_number [-verbose] [-force]

Example

$ srvctl modify oc4j -rmiport 5385

---

srvctl modify ons

Modifies the ports used by the Oracle Notification Service daemon that is registered with Oracle Restart.

Syntax

srvctl modify ons [-onslocalprt ons_local_port] [-onsremoteport ons_remote_port] [-emport em_port] [-remoteservers host[:port][,host[:port]][...]] [-verbose]

---

srvctl modify scan

Modifies the number of SCAN VIPs to match the number of IP addresses returned by looking up the scan_name you specify in DNS. You use this command when DNS was modified to add, change, or remove IP addresses, and now you must adjust the Oracle Clusterware resource configuration to match.

Syntax

srvctl modify scan -scanname scan_name

Example

Assume your system currently has a SCAN named scan_name1, and it resolves to a single IP address in DNS. If you modify the SCAN scan_name1 in DNS to resolve to three IP addresses, then use the following command to create the additional SCAN VIP resources:

$ srvctl modify scan -scanname scan_name1

---

srvctl modify scan_listener

Modifies the SCAN listener to match SCAN VIP's or modifies the SCAN listener endpoints or service registration restrictions.

Syntax

srvctl modify scan_listener {-update | -endpoints [TCP:]port[/IPC:key] [/NMP:pipe_name] [/TCPS:s_port][/SDP:port] } [-invitednodes node_list] [-invitedsubnets subnet_list]

Example

Assume your system currently has a SCAN named scan_name1, and you recently modified the DNS entry to resolve to three IP addresses instead of one. After running the srvctl modify scan command to create additional SCAN VIP resources, use the following command to create Oracle Clusterware resources for the additional two SCAN listeners to go with the two additional SCAN VIPs:

$ srvctl modify scan_listener -update

---

srvctl modify service

This command supports some online modifications to the service, such as:

Moving a service member from one instance to another

Performing online changes to service attributes from DBMS_SERVICE (for example, failover delay, runtime load balancing goal, and so on)

Adding a new preferred or available instance

Removing preferred or available instances for a service

syntax:

To move a service from one instance to another:

srvctl modify service -db db_unique_name -service service_name -oldinst old_instance_name -newinst new_instance_name [-force]

Note:

This form of the command is only available with Oracle Clusterware.

To change an available instance to a preferred instance for a service:

srvctl modify service -db db_unique_name -service service_name -available avail_inst_name -toprefer [-force]

To change the available and preferred status for multiple instances:

srvctl modify service -db db_unique_name -service service_name -modifyconfig -preferred "preferred_list" [-available "available_list"] [-force]

To modify other service attributes or to modify a service for Oracle Clusterware:

srvctl modify service -db db_unique_name -service service_name [-eval] [-serverpool pool_name] [-cardinality {UNIFORM|SINGLETON}] [-pqservice pqsvc_name] [-pqpool pq_pool_list] [-tafpolicy {BASIC|NONE}] [-edition edition_name] [-role "[PRIMARY][,PHYSICAL_STANDBY][,LOGICAL_STANDBY][,SNAPSHOT_STANDBY]"] [-notification {TRUE|FALSE}] [-dtp {TRUE|FALSE}] [-clbgoal {SHORT|LONG}] [-rlbgoal {NONE|SERVICE_TIME|THROUGHPUT}] [-failovertype {NONE|SESSION|SELECT|TRANSACTION}] [-failovermethod {NONE|BASIC}] [-failoverretry failover_retries] [-failoverdelay failover_delay] [-policy {AUTOMATIC | MANUAL}] [-pdb pluggable_database] [-sql_translation_profile profile_name] [-commit_outcome {TRUE|FALSE}] [-retention retention_time] [-replay_init_time replay_initiation_time] [-session_state {STATIC|DYNAMIC}] [-global_override] [-verbose] [-force]

Examples

An example of moving a service member from one instance to another is:

$ srvctl modify service -db crm -service crm -oldinst crm1 -newinst crm2

An example of changing an available instance to a preferred instance is:

$ srvctl modify service -db crm -service crm -available crm1 -toprefer

The following command exchanges a preferred and available instance:

$ srvctl modify service -db crm -service crm -modifyconfig -preferred "crm1" \ -available "crm2"

---

srvctl modify srvpool

Modifies a server pool in a cluster. If minimum size, maximum size, and importance are numerically increased, then the CRS daemon may attempt to reassign servers to this server pool, if by resizing other server pools have comparatively lower minimum size and importance, to satisfy new sizes of this server pool.

Syntax

srvctl modify srvpool -serverpool pool_name [-eval] [-importance importance] [-min min_size] [-max max_size] [-servers "server_list"] [-category "server_category"] [-verbose] [-force]

Example

The following example changes the importance rank to 0, the minimum size to 2, and the maximum size to 4 for the server pool srvpool1:

$ srvctl modify srvpool -serverpool srvpool1 -importance 0 -min 2 -max 4

---

srvctl modify vip

Modifies IP address type but you can also use it to modify just the IP address.

Syntax

srvctl modify vip -node node_name -address {VIP_name|ip}/netmask[/if1[|if2|...]] [-netnum network_number] [-verbose]

Example

The following example adds an IPv4 address to a VIP, if one does not already exist. If the VIP has an IPv4 address, then it is replaced with the new network specification.

srvctl modify vip -node node7 -address 192.168.16.17/255.255.255.0 -netnum 2

---

predict

The predict command helps you evaluate the consequences of a resource failure. This command does not make any modifications to the system.

---

srvctl predict asm

Predicts the consequences of ASM failure.

Syntax

srvctl predict asm [-node node_name] [-verbose]

Examples

$ srvctl predict asm -node crmnode2

vahidusefzadeh@gmail.com

---

srvctl predict database

The srvctl predict database command predicts what happens if the specified database fails.

Syntax

srvctl predict database -db db_unique_name [-verbose]

Example

srvctl predict database -db racdb

---

srvctl predict diskgroup

Predicts the consequences of and Oracle ASM disk group failure.

Syntax

srvctl predict diskgroup -diskgroup diskgroup_name [-verbose]

Example

$ srvctl predict diskgroup -diskgroup data

---

srvctl predict filesystem

Predicts the consequences of file system failure.

Syntax

srvctl predict filesystem -device volume_device_name [-verbose]

Example

srvctl predict filesystem -device /dev/asm/volume1-123

---

srvctl predict listener

Predicts the consequences of listener failure.

Syntax

srvctl predict listener listener_name [-verbose]

Example

$ srvctl predict listener -listener NODE3_CRMAPP_LISTENER

---

srvctl predict network

Predicts the consequences of network failure.

Syntax

srvctl predict network [-netnum network_number [-verbose]

Example

$ srvctl predict network -netnum 2

---

srvctl predict oc4j

Predicts the consequences of Oracle Container for Java (OC4J) failure.

Syntax

srvctl predict oc4j [-verbose]

Example

$ srvctl predict oc4j

---

srvctl predict scan

Predicts the consequences of SCAN failure.

Syntax

srvctl predict scan -scannumber ordinal_number [-verbose]

Example

$ srvctl predict scan -scannumber 1 -verbose

---

srvctl predict scan_listener

Predicts the consequences of SCAN listener failure.

Syntax

srvctl predict scan_listener -scannumber ordinal_number [-verbose]

Example

$ srvctl predict scan_listener -scannumber 1

---

srvctl predict service

Predicts the consequences of service failure.

Syntax

srvctl predict service -db db_unique_name -service service_name [-verbose]

Example

$ srvctl predict service -db racdb -service "crm"

---

srvctl predict vip

Predicts the consequences of VIP failure.

Syntax

srvctl predict vip [-vip vip_name] [-verbose]

Example

$ srvctl predict vip -vip racdb1_vip

---

relocate

The relocate command causes the specified object to run on a different node. The specified object must be running already.

The relocation of the object is temporary until you modify the configuration. The previously described modify command permanently changes the configuration.

---

srvctl relocate cvu

Temporarily relocates the CVU to another node in a cluster.

Syntax

srvctl relocate cvu [-node node_name]

Example

$ srvctl relocate cvu -node crmnode2

---

srvctl relocate database

The srvctl relocate database command initiates the relocation of an Oracle RAC One Node database from one node to another node. This command also cleans up after a failed relocation.

The srvctl relocate database command can only be used for relocating Oracle RAC One Node databases.

Syntax

srvctl relocate database -db db_unique_name [-node target_node] [-timeout timeout] [-verbose]

srvctl relocate database -db db_unique_name -abort [-revert] [-verbose]

Example

The following example relocates an administrator-managed Oracle RAC One Node database named rac1 to a server called node7.

srvctl relocate database -db rac1 -node node7

---

srvctl relocate gns

Relocates GNS from its current hosting node to another node within the cluster.

Syntax

srvctl relocate gns [-node node_name] [-verbose]

Example

srvctl relocate gns -node node1

---

srvctl relocate havip

Relocates a highly available VIP (HAVIP) (used for highly available NFS exports) to another node in a cluster.

Syntax

srvctl relocate havip -id havip_name [-node node_name] [-force]

Example

$ srvctl relocate havip -id hrexports -node node3

---

srvctl relocate mgmtdb

Relocates the management database (CHM repository) resource from one node of the cluster to another.

Syntax

srvctl relocate mgmtdb [-node node_name]

Example

$ srvctl relocate mgmtdb -node crsNode2

---

srvctl relocate oc4j

Relocates an OC4J instance from its current hosting node to another node within the cluster.

Syntax

srvctl relocate oc4j [-node node_name] [-verbose]

Example

$ srvctl relocate oc4j -node crsNode01 -verbose

---

srvctl relocate scan

Relocates a specific SCAN VIP from its current hosting node to another node within the cluster.

Syntax

srvctl relocate scan -scannumber ordinal_number [-node node_name]

Example

$ srvctl relocate scan -scannumber 1 -node node1

---

srvctl relocate scan_listener

Relocates a specific SCAN listener from its current hosting node to another node within the cluster.

Syntax

srvctl relocate scan_listener -scannumber ordinal_number [-node node_name]

Example

$ srvctl relocate scan_listener -scannumber 3

---

srvctl relocate server

vahidusefzadeh@gmail.com

Relocates servers to a server pool in the cluster.

Syntax

srvctl relocate server -servers "server_name_list" -serverpool pool_name [-eval] [-force]

Example

$ srvctl relocate server -servers "server1, server2" -serverpool sp3

---

srvctl relocate service

Relocates the named service names from one named instance to another named instance. The srvctl relocate command works on only one source instance and one target instance at a time, relocating a service from a single source instance to a single target instance. The target instance must be on the preferred or available list for the service.

Syntax

srvctl relocate service -db db_unique_name -service service_name [-eval] -currentnode source_node -targetnode target_node [-force] srvctl relocate service -db db_unique_name -service service_name [-eval] -oldinst old_instance_name -newinst new_instance_name [-force]

Example

$ srvctl relocate service -db crm -service crm -oldinst crm1 -newinst crm3

---

srvctl relocate vip

Relocates a specific VIP from its current hosting node to another node within the cluster.

Syntax

srvctl relocate vip -vip vip_name [-node node_name] [-force] [-verbose]

Example

$ srvctl relocate vip -vip vip1 -node node3

---

remove

Removes the configuration information for the specified target from Oracle Clusterware. Environment settings for the object are also removed. Using this command does not destroy the specified target.

Use the remove verb to remove the associated resource from the management of Oracle Clusterware or Oracle Restart. Depending on the noun used, you can remove databases, services, nodeapps, Oracle ASM, Oracle Notification Service, and listeners.

If you do not use the force parameter (-force), then Oracle Clusterware or Oracle Restart prompts you to confirm whether to proceed. If you use -force, then the remove operation proceeds without prompting and continues processing even when it encounters errors. Even when the Oracle Clusterware resources cannot be removed, the OCR configuration is removed, so that the object now appears not to exist, but there are still Oracle Clusterware resources. Use the force parameter (-force) with extreme caution because this could result in an inconsistent OCR.

To use the remove verb, you must first stop the node applications, database, instance, or service for which you are specifying srvctl remove. Oracle recommends that you perform a disable operation before using this command, but this is not required. You must stop the target object before running thesrvctl remove command. See the stop command.

---

srvctl remove asm

Removes the Oracle ASM resource from Oracle Clusterware management.

Syntax

srvctl remove asm [-proxy] [-force]

Example

$ srvctl remove asm -force

---

srvctl remove cvu

Removes CVU from an Oracle Clusterware configuration.

Syntax

srvctl remove cvu [-force]

Example

$ srvctl remove cvu -force

---

srvctl remove database

Removes a database configuration.

Syntax

srvctl remove database -db db_unique_name [-force] [-noprompt] [-verbose]

Example

$ srvctl remove database -db crm

---

srvctl remove diskgroup

Removes a specific Oracle ASM disk group resource from Oracle Clusterware or Oracle Restart.

Syntax

srvctl remove diskgroup -diskgroup diskgroup_name [-force]

Example

$ srvctl remove diskgroup -diskgroup DG1 -force

---

srvctl remove exportfs

Removes the specified export file system configuration.

Syntax

srvctl remove exportfs -name exportfs_name [-force]

Examples

To remove the stopped export file system named export1:

srvctl remove exportfs -name export1

To remove a running export, leaving it exported:

srvctl remove exportfs -name export1 -force

---

srvctl remove filesystem

Removes a specific file system resource from the cluster.

Syntax

srvctl remove filesystem -device volume_device_name [-force]

Example

srvctl remove filesystem -device /dev/asm/racvol1

---

srvctl remove gns

Removes GNS from the cluster.

Syntax

srvctl remove gns [-force] [-verbose]

Example

$ srvctl remove gns

---

srvctl remove havip

Removes a highly available VIP (HAVIP) (used for highly available NFS exports).

Syntax

srvctl remove havip -id havip_name [-force]

Example

srvctl remove havip -id hrexports

---

srvctl remove instance

Removes the configurations for an instance of an administrator-managed database. To remove the configurations of a policy-managed database, you must shrink the size of the server pool with the srvctl modify srvpool command.

If you use the -force parameter, then any services running on the instance stop. Oracle recommends that you reconfigure services to not use the instance to be removed as a preferred or available instance before removing the instance.

Notes:

This command is only available with Oracle Clusterware and Oracle RAC.

If you attempt to use this command on an Oracle RAC One Node database, then the command returns an error stating that cannot remove the instance except by removing the database.

Syntax

srvctl remove instance -db db_unique_name -instance instance_name [-noprompt] [-force]

Example

$ srvctl remove instance -db crm -instance crm01

---

srvctl remove listener

Removes the configuration of a specific listener or all listeners from Oracle Clusterware or Oracle Restart.

Syntax

srvctl remove listener [-listener listener_name | -all] [-force]

Example

$ srvctl remove listener -listener lsnr01

---

srvctl remove mgmtdb

Removes the management database (CHM repository) from Oracle Clusterware management.

Syntax

srvctl remove mgmtdb [-force] [-noprompt] [-verbose]

Example

$ srvctl remove mgmtdb -noprompt

---

srvctl remove mgmtlsnr

Use this command to remove the management listener resource (for CHM) from Oracle Clusterware.

Syntax

srvctl remove mgmtlsnr [-force]

Example

srvctl remove mgmtlsnr

---

srvctl remove network

Removes the network configuration. You must have full administrative privileges to run this command. On Linux and UNIX systems, you must be logged in asroot and on Windows systems, you must be logged in as a user with Administrator privileges.

Syntax

srvctl remove network {-netnum network_number | -all} [-force] [-verbose]

Example

srvctl remove network -netnum 3

---

srvctl remove nodeapps

Removes the node application configuration.

Syntax

srvctl remove nodeapps [-force] [-noprompt] [-verbose]

Example

srvctl remove nodeapps

---

srvctl remove oc4j

Removes the OC4J instance from the Oracle Clusterware configuration.

Syntax

srvctl remove oc4j [-force] [-verbose]

---

srvctl remove ons

Removes Oracle Notification Service from the Oracle Grid Infrastructure home.

Note:

This command is only available with Oracle Restart.

Syntax

srvctl remove ons [-force] [-verbose]

---

srvctl remove scan

Removes Oracle Clusterware resources from all SCAN VIPs.

Syntax

srvctl remove scan [-force] [-noprompt]

Example

$ srvctl remove scan -force

---

srvctl remove scan_listener

Removes Oracle Clusterware resources from all SCAN listeners.

Syntax

srvctl remove scan_listener [-force] [-noprompt]

Example

$ srvctl remove scan_listener -force

---

srvctl remove service

Removes the configuration for a service.

Syntax

srvctl remove service -db db_unique_name -service service_name [-instance instance_name] [-global_override] [-force]

Examples

This command removes the sales service from all instances of the clustered database named crm:

$ srvctl remove service -db crm -service sales

The following example removes the sales service from a specific instance of the crm clustered database:

$ srvctl remove service -db crm -service sales -instance crm02

---

vahidusefzadeh@gmail.com

srvctl remove srvpool

Removes a specific server pool. If there are databases or services that depend upon this server pool, then those resources are removed from the server pool first so that the remove server pool operation succeeds.

Syntax

srvctl remove srvpool -serverpool pool_name [-eval] [-verbose]

Example

$ srvctl remove srvpool -serverpool srvpool1

---

srvctl remove vip

Removes specific VIPs.

Syntax

srvctl remove vip -vip "vip_name_list" [-force] [-noprompt] [-verbose]

Example

$ srvctl remove vip -vip "vip1,vip2,vip3" -force -noprompt -verbose

---

srvctl remove volume

Removes a specific volume.

Syntax

srvctl remove volume -volume volume_name -diskgroup disk_group_name [-force]

srvctl remove volume -device volume_device [-force]

Example

$ srvctl remove volume -volume VOLUME1 -diskgroup DATA

---

vahidusefzadeh@gmail.com

---

setenv

The setenv command sets values for the environment in the configuration file. Use setenv to set environment variables—items such as language or TNS_ADMIN—for Oracle Clusterware that you would typically set in your profile or session when you manage this database or database instance.

The unsetenv command unsets values for the environment in the configuration file.

---

srvctl setenv asm

Administers Oracle ASM environment configurations.

Syntax

```
srvctl setenv asm -envs "name=val[,name=val][...]"

srvctl setenv asm -env "name=val"
```

Example

```
$ srvctl setenv asm -env "LANG=en"
```

---

srvctl setenv database

Administers cluster database environment configurations.

Syntax

```
srvctl setenv database -db db_unique_name -envs "name=val[,name=val][...]"

srvctl setenv database -db db_unique_name -env "name=val"
```

Example

```
$ srvctl setenv database -db crm -env LANG=en
```

---

srvctl setenv listener

vahidusefzadeh@gmail.com

Administers listener environment configurations.

Syntax

srvctl setenv listener [-listener listener_name] -envs "name=val[,name=val][...]"

srvctl setenv listener [-listener listener_name] -env "name=val"

Example

$ srvctl setenv listener -env "LANG=en"

---

srvctl setenv mgmtdb

Administers the environment configuration for the management database (CHM repository).

Syntax

srvctl setenv mgmtdb -envs "name=val[,name=val][...]"

srvctl setenv mgmtdb -env "name=val"

Example

$ srvctl setenv mgmtdb -env LANG=en

---

srvctl setenv mgmtlsnr

Administers the environment configuration for the management listener resource (for CHM).

Syntax

srvctl setenv mgmtlsnr -envs "name=val[,name=val][...]"

srvctl setenv mgmtlsnr -env "name=val"

Example

$ srvctl setenv mgmtlsnr -env LANG=en

---

vahidusefzadeh@gmail.com

srvctl setenv nodeapps

Sets the environment variables for the node application configurations.

Syntax

srvctl setenv nodeapps {-namevals "name=val[,name=val][...]" | -nameval "name=val"} [-viponly] [-gsdonly] [-onsonly] [-verbose]

Example

$ srvctl setenv nodeapps -env "CLASSPATH=/usr/local/jdk/jre/rt.jar" -verbose

---

srvctl setenv vip

Administers cluster VIP environment configurations.

Syntax

srvctl setenv vip -vip vip_name {-envs "name=val[,name=val,...]" | -env "name=val"} [-verbose]

Example

The following example sets the language environment configuration for a cluster VIP:

$ srvctl setenv vip -vip crm1-vip -env "LANG=en"

---

start

Starts Oracle Restart or Oracle Clusterware enabled, non-running applications for the database, all or named instances, all or named service names, or node-level applications. For the start command, and for other operations that use a connect string, if you do not provide a connect string, SRVCTL uses /as sysdbato perform the operation. To run such operations, the owner of the oracle binary executables must be a member of the OSDBA group, and users running the commands must also be in the OSDBA group.

---

srvctl start asm

Starts an Oracle ASM instance.

Notes:

To manage Oracle ASM on Oracle Database 12c installations, use the SRVCTL binary in the Oracle Grid Infrastructure home for a cluster (Grid home). If you have Oracle RAC or Oracle Database installed, then you cannot use the SRVCTL binary in the database home to manage Oracle ASM.

Syntax

srvctl start asm [-proxy] [-node node_name [-startoption start_options]]

Examples

An example of this command to start an Oracle ASM instance on a single node of a cluster is:

$ srvctl start asm -node crmnode1

An example to start an Oracle ASM instance on all nodes in the cluster, or for a noncluster database, is:

$ srvctl start asm

---

srvctl start cvu

Starts the CVU resource on one node in a cluster. If you specify a node name, then CVU starts on that node.

Syntax

srvctl start cvu [-node node_name]

Examples

An example of this command to start CVU on a single node of a cluster is:

$ srvctl start cvu -node crmnode1

---

srvctl start database

Starts a cluster database and its enabled instances and all listeners on nodes with database instances. You can disable listeners that should not be started.

vahidusefzadeh@gmail.com

Syntax

srvctl start database -db db_unique_name [-eval] [-startoption start_options] [-node node_name]

Example

$ srvctl start database -db crm -startoption 'read only'

---

srvctl start diskgroup

Starta a specific disk group resource on a number of specified nodes.

Syntax

srvctl start diskgroup -diskgroup diskgrp_name [-node "node_list"]

Example

$ srvctl start diskgroup -diskgroup diskgroup1 -node "mynode1,mynode2"

---

srvctl start exportfs

Starts an export file system configuration in Oracle Clusterware.

Syntax

srvctl start exportfs {-name unique_name | -id havip_name}

Example

$ srvctl start exportfs -name export1

---

srvctl start filesystem

Starts (mounts) the file system resource.

Syntax

srvctl start filesystem -device volume_device [-node node_name]

Examples

To start a file system on all configured nodes in the cluster:

```
$ srvctl start filesystem -device /dev/asm/data_db1-68
```

To start the file system on node1:

```
$ srvctl start filesystem -device /dev/asm/data_db1-68 -node node1
```

---

srvctl start gns

Starts GNS on a specific node.

Syntax

```
srvctl start gns [-loglevel log_level] [-node node_name] [-verbose]
```

Example

An example of this command to start the GNS on the cluster node named crmnode1 is:

```
$ srvctl start gns -node crmnode1
```

---

srvctl start havip

Starts a specific highly available VIP (HAVIP) (used for highly available NFS exports) on a specific node.

Syntax

```
srvctl start havip -id havip_name [-node node_name]
```

Example

```
# srvctl start havip -id hrexports -node myNode1
```

---

srvctl start home

Starts all the Oracle Restart-managed or Oracle Clusterware-managed resources on the specified Oracle home.

Syntax

srvctl start home -oraclehome Oracle_home -statefile state_file -node node_name

Example

$ srvctl start home -oraclehome /u01/app/oracle/product/12.1.0/db_1 -statefile ~/state.txt -node node1

---

srvctl start instance

Starts instances in the cluster database and all listeners on nodes with database instances. You can disable listeners that should not be started.

Syntax

srvctl start instance -db db_unique_name -node node_name [-instance "instance_name"] [-startoption start_options]

srvctl start instance -db db_unique_name -instance "inst_name_list" [-startoption start_options]

Example

An example of starting an instance for a policy-managed database is:

$ srvctl start instance -db crm -node node2

An example of starting instances for an administrator-managed database is:

$ srvctl start instance -db crm -instance "crm2,crm3"

---

srvctl start listener

Starts the default listener on the specified node_name, or starts the specified listener on all nodes that are registered with Oracle Clusterware or on the given node.

Syntax

srvctl start listener [-node node_name] [-listener listener_name]

Example

$ srvctl start listener -listener LISTENER_LEAF

---

srvctl start mgmtdb

Starts the management database (CHM repository) resource.

Syntax

srvctl start mgmtdb [-startoption start_options] [-node node_name]

Examples

An example of this command to start the management database on the crmnode1 node of the cluster is:

$ srvctl start mgmtdb -node crmnode1

---

srvctl start mgmtlsnr

Starts the management listener resource (for CHM).

Syntax

srvctl start mgmtlsnr [-node node_name]

Examples

For example, to start the management listener on the crmnode1 node:

$ srvctl start mgmtlsnr -node crmnode1

---

srvctl start nodeapps

Starts node-level applications on a node or all nodes in the cluster.

Syntax

srvctl start nodeapps [-node node_name] [-gsdonly] [-adminhelper] [-verbose]

Example

$ srvctl start nodeapps

---

srvctl start oc4j

Starts the OC4J instance.

Syntax

srvctl start ocj4 [-verbose]

---

srvctl start ons

Starts the Oracle Notification Service daemon.

Note:

This command is only available with Oracle Restart.

Syntax

srvctl start ons [-verbose]

There is only one parameter for this command, -verbose, which is used to indicate that verbose output should be displayed.

---

srvctl start scan

Starts all SCAN VIPs, by default, or a specific SCAN VIP, on all nodes or a specific node in the cluster.

Syntax

srvctl start scan [-scannumber ordinal_number] [-node node_name]

Example

To start the SCAN VIP identified by the ordinal number 1 on the crm1 node, use the following command:

$ srvctl start scan -scannumber 1 -node crm1

vahidusefzadeh@gmail.com

---

srvctl start scan_listener

Starts all SCAN listeners, by default, or a specific listener on all nodes or a specific node in the cluster.

Syntax

srvctl start scan_listener [-node node_name] [-scannumber ordinal_number]

Example

$ srvctl start scan_listener -scannumber 1

---

srvctl start service

Starts a service or multiple services on the specified instance. The srvctl start service command will fail if you attempt to start a service on an instance if that service is already running on its maximum number of instances, that is, its number of preferred instances. You may move a service or change the status of a service on an instance with the srvctl modify service and srvctl relocate service commands described later in this appendix.

Syntax

srvctl start service -db db_unique_name [-eval] [-service "services_list" [-node node_name | -instance instance_name | -serverpool pool_name | -global_override] [-startoption start_options] [-verbose]

Examples

$ srvctl start service -db crm -service crm

The following example starts a named service on a specified instance:

$ srvctl start service -db crm -service crm -instance crm2

---

srvctl start vip

Starts a specific VIP or a VIP on a specific node.

Syntax

srvctl start vip {-node node_name | -vip vip_name } [-verbose]

Example

$ srvctl start vip -vip crm1-vip -verbose

---

srvctl start volume

Starts a specific, enabled volume.

Syntax

srvctl start volume {-volume volume_name -diskgroup disk_group_name | -device volume_device} [-node node_list]

Example

The following example starts a volume named VOLUME1 that resides in a disk group named DATA:

$ srvctl start volume -volume VOLUME1 -diskgroup DATA

---

srvctl status asm

Displays the status of an Oracle ASM instance.

Syntax

srvctl status asm [-proxy] [-node node_name] [-detail] [-verbose]

Example

$ srvctl status asm -node crmnode1 -detail

---

srvctl status cvu

Displays the current state of the CVU resource on one node in a cluster. If you specify a node name, then the command checks CVU status on that node.

Syntax

srvctl status cvu [-node node_name]

Examples

$ srvctl status cvu -node crmnode1

---

srvctl status database

Displays the status of instances and their services, and where the instances are running.

If you run this command on an Oracle RAC One Node database, then the output shows the status of any online database relocation (active, failed, or inactive), and the source and destination nodes of the relocation.

Syntax

srvctl status database -db db_unique_name [-force] [-verbose]

Example

$ srvctl status database -db crm -verbose

---

srvctl status diskgroup

Displays the status of a specific disk group on a number of specified nodes.

Syntax

srvctl status diskgroup -diskgroup diskgroup_name [-node "node_list"] [-detail] [-verbose]

Example

$ srvctl status diskgroup -diskgroup dgrp1 -node "mynode1,mynode2" -detail

---

srvctl status exportfs

Displays the status of an export file system configuration.

Syntax

srvctl status exportfs [-name unique_name | -id havip_name]

Example

$ srvctl status exportfs

---

srvctl status filesystem

Displays the status of the file system resource.

Syntax

srvctl status filesystem [-device volume_device] [-verbose]

Examples

This command displays output similar to the following, depending on whether you specify a device name:

If you specify a device name:

$ srvctl status filesystem -device /dev/asm/racvol_1

ACFS file system is not mounted on node1

ACFS file system is not mounted on node2

If you do not specify a device name:

$ srvctl status filesystem

ACFS file system is not running

ACFS file system is running on node1,node3

Note:

In the preceding examples, the file system is Oracle ACFS. If you are using other file systems, then they will display as EXT3 or EXT4.

---

srvctl status gns

Displays the current state of GNS.

Syntax

srvctl status gns [-node node_name] [-verbose]

---

srvctl status havip

Displays the status of all highly available VIPs (HAVIPs) (used for highly available NFS exports) in a cluster or one particular highly available VIP.

Syntax

srvctl status havip [-id havip_name]

Example

$ srvctl status havip

HAVIP ora.ha1.havip is enabled

HAVIP ora.ha1.havip is not running

---

srvctl status home

Displays the status of all the Oracle Restart-managed or Oracle Clusterware-managed resources for the specified Oracle home.

Syntax

srvctl status home -oraclehome Oracle_home -statefile state_file -node node_name

Example

$ srvctl status home -oraclehome /u01/app/oracle/product/12.1/dbhome_1 -statefile

~/state.txt -node stvm12

The preceding command returns output similar to the following:

Database cdb1 is running on node stvm12

---

vahidusefzadeh@gmail.com

srvctl status instance

Displays the status of instances.

Syntax

srvctl status instance -db db_unique_name {-node node_name | -instance "instance_name_list"} [-force] [-verbose]

Example

$ srvctl status instance -db crm -instance "crm1,crm2" -verbose

---

srvctl status listener

Displays the status of listener resources.

Syntax

srvctl status listener [-listener listener_name] [-node node_name] [-verbose]

Example

$ srvctl status listener -node node2

---

srvctl status mgmtdb

Displays the current state of the management database (CHM repository) resource.

Syntax

srvctl status mgmtdb [-verbose]

Examples

$ srvctl status mgmtdb

---

srvctl status mgmtlsnr

Displays the status of the management listener resource (for CHM).

Syntax

srvctl status mgmtlsnr [-node node_name] [-verbose]

Example

$ srvctl status mgmtlsnr -node node2

---

srvctl status nodeapps

Displays the status of node applications.

Syntax

srvctl status nodeapps [-node node_name]

---

srvctl status oc4j

Determines which node is running the Oracle Database QoS Management server.

Syntax

srvctl status oc4j [-node node_name] [-verbose]

---

srvctl status ons

Displays the current state of the Oracle Notification Service daemon.

Note:

This command is only available with Oracle Restart.

Syntax

srvctl status ons

---

vahidusefzadeh@gmail.com

srvctl status scan

Displays the status for all SCAN VIPs, by default, or a specific SCAN VIP.

Syntax

srvctl status scan [-scannumber ordinal_number] [-verbose]

Example

$ srvctl status scan -scannumber 1

---

srvctl status scan_listener

Displays the status for all SCAN listeners, by default, or a specific listener.

Syntax

srvctl status scan_listener [-scannumber ordinal_number] [-verbose]

Example

$ srvctl status scan_listener -scannumber 1

---

srvctl status server

Displays the current state of named servers.

Syntax

srvctl status server -server "server_name_list" [-detail]

Example

$ srvctl status server -server "server11" -detail

---

srvctl status service

Displays the status of a service.

For Oracle RAC One Node databases, if there is an online database relocation in process, then the srvctl status service command displays the source and destination nodes and the status of the relocation, whether it is active or failed.

Syntax

srvctl status service -db db_unique_name [-service "service_name_list"] [-force] [-verbose]

Example

$ srvctl status service -db crm -service "crm" -verbose

---

srvctl status srvpool

Displays server pool names, number of servers in server pools, and, optionally, the names of the servers in the server pools.

Syntax

srvctl status srvpool [-serverpool pool_name] [-detail]

Example

$ srvctl status srvpool -serverpool srvpool1 -detail

---

srvctl status vip

Displays status for a specific VIP or a VIP on a specific node.

Syntax

srvctl status vip {-node node_name | -vip vip_name} [-verbose]

Example

$ srvctl status vip -vip node1-vip

---

srvctl status volume

Displays the status of a specific volume or all volumes.

Syntax

srvctl status volume [-device volume_device] [-volume volume_name] [-diskgroup disk_group_name] [-node "node_list"]

Example

$ srvctl status volume –volume vol1

---

srvctl stop asm

Stops an Oracle ASM instance.

Syntax

srvctl stop asm [-proxy] [-node node_name] [-stopoption stop_options] [-force]

Example

$ srvctl stop asm -node crmnode1 -stopoption IMMEDIATE

---

srvctl stop cvu

Stops the Cluster Verification Utility (CVU) that is either in the running or starting state.

Syntax

srvctl stop cvu [-force]

Example

$ srvctl stop cvu

---

srvctl stop database

vahidusefzadeh@gmail.com

Stops a database, its instances, and its services. When the database later restarts, services with AUTOMATIC management start automatically but services withMANUAL management policy must be started manually.

Syntax

srvctl stop database -db db_unique_name [-stopoption stop_options] [-eval][-force] [-verbose]

Example

$ srvctl stop database -db crm -stopoption NORMAL

---

srvctl stop diskgroup

Use this command to stop a specific disk group resource on a number of specified nodes.

Syntax

srvctl stop diskgroup -diskgroup diskgroup_name [-node "node_list"] [-force]

Example

$ srvctl stop diskgroup -diskgroup diskgroup1 -node "mynode1,mynode2" -force

---

srvctl stop exportfs

Use this command to stop an export file system configuration managed by Oracle Clusterware.

Syntax

srvctl stop exportfs {-name unique_name | -id havip_name} [-force]

Example

srvctl stop exportfs -name export1

---

srvctl stop filesystem

Use this command to stop (unmount) the Oracle ACFS file system or generic file system.

Syntax

srvctl stop filesystem -device volume_device [-node node_name] [-force]

Example

$ srvctl stop filesystem -device /dev/asm/racvol_1 -force

---

srvctl stop gns

Use this command to stop GNS in the cluster.

Syntax

srvctl stop gns [-node node_name] [-verbose] [-force]

Example

$ srvctl stop gns

---

srvctl stop havip

Stops the highly available VIPs (HAVIPs) (used for highly available NFS exports) on a specific node.

Syntax

srvctl stop havip -id havip_name [-node node_name] [-force]

Example

srvctl stop havip -id hrexports -node myNode1

---

srvctl stop home

Stops all the Oracle Restart-managed or Oracle Clusterware-managed resources that run from the specified Oracle home.

Syntax

srvctl stop home -oraclehome Oracle_home -statefile state_file -node node_name [-stopoption stop_options] [-force]

Example

$ srvctl stop home -oraclehome /u01/app/oracle/product/12.1.0/db_1 -statefile

~/state.txt

---

srvctl stop instance

Stops instances and stops any services running on specified instances, unless you specify the -force parameter. If you specify -force, then the services fail over to an available instance when the instance stops.

Syntax

srvctl stop instance -db db_unique_name {-node node_name | -instance "instance_name_list"} [-stopoption stop_options] [-force]

Example

An example of stopping an instance in a policy-managed database is:

$ srvctl stop instance -db crm -node node1

An example of stopping an instance in an administrator-managed database is:

$ srvctl stop instance -db crm -instance "crm1"

---

srvctl stop listener

Stops the default listener or a specific listener on all nodes or the specified node.

This command can also be used to stop a listener on a noncluster database from the noncluster database home. However, SRVCTL does not accept the -nodeparameter when run from a noncluster database home.

Syntax

srvctl stop listener [-listener listener_name] [-node node_name] [-force]

vahidusefzadeh@gmail.com

Example

$ srvctl stop listener -node mynode1

---

srvctl stop mgmtdb

Stops the management database (CHM repository) resource.

Syntax

srvctl stop mgmtdb [-stopoption stop_options] [-force]

Example

$ srvctl stop mgmtdb -stopoption NORMAL

---

srvctl stop mgmtlsnr

Stops the management listener resource (for CHM) on all nodes or the specified node.

Syntax

srvctl stop mgmtlsnr [-node node_name] [-force]

Example

$ srvctl stop mgmtlsnr -node mynode1

---

srvctl stop nodeapps

Stops node-level applications on a node in the cluster.

Syntax

srvctl stop nodeapps [-node node_name] [-gsdonly] [-adminhelper] [-force] [-relocate] [-verbose]

Example

$ srvctl stop nodeapps

---

srvctl stop oc4j

Stops the OC4J instance that is in the running or starting state.

Syntax

srvctl stop oc4j [-force] [-verbose]

Example

$ srvctl stop oc4j -force -verbose

---

srvctl stop ons

Stops the Oracle Notification Service daemon.

Note:

This command is only available with Oracle Restart.

Syntax

srvctl stop ons [-verbose]

Example

$ srvctl stop ons -verbose

---

srvctl stop scan

Stops all SCAN VIPs, by default, that are running or in starting state, or stops a specific SCAN VIP identified by ordinal_number.

Syntax

srvctl stop scan [-scannumber ordinal_number] [-force]

Example

```
$ srvctl stop scan -scannumber 1
```

---

srvctl stop scan_listener

Stops all SCAN listeners, by default, that are in a running or starting state, or a specific listener identified by ordinal_number.

Syntax

```
srvctl stop scan_listener [-scannumber ordinal_number] [-force]
```

Example

```
$ srvctl stop scan_listener -scannumber 1
```

---

srvctl stop service

Stops one or more services globally across the cluster database, or on the specified instance.

Syntax

```
srvctl stop service -db db_unique_name [-service "services_list"   [-node node_name | -instance instance_name | -serverpool pool_name |   -global_override]] [-eval] [-force]
```

Examples

The following example stops a service for all cluster database instances:

```
$ srvctl stop service -db crm -service "crm"
```

The following example stops a service on a specified instance:

```
$ srvctl stop service -db crm -service "crm" -instance crm2
```

---

srvctl stop vip

Stops a specific VIP or all VIPs on a specific node, including any VIPs that were relocated due to a failover.

Syntax

srvctl stop vip {-node node_name | -vip vip_name} [-force] [-relocate] [-verbose]

Example

$ srvctl stop vip -node mynode1 -verbose

---

srvctl stop volume

Stops a specific, running volume.

Syntax

srvctl stop volume {-volume volume_name -diskgroup disk_group_name | -device volume_device} [-node "node_list"]

Example

$ srvctl stop volume -volume VOLUME1 -diskgroup DATA

---

srvctl unsetenv asm

Unsets the Oracle ASM environment configurations.

Syntax

srvctl unsetenv asm -envs "name_list"

Example

$ srvctl unsetenv asm -envs "CLASSPATH"

---

srvctl unsetenv database

Unsets the cluster database environment configurations.

Syntax

srvctl unsetenv database -db db_unique_name -envs "name_list"

Example

$ srvctl unsetenv database -db crm -envs "CLASSPATH,LANG"

---

srvctl unsetenv listener

Unsets the environment configuration for a listener.

Syntax

srvctl unsetenv listener [-listener listener_name] -envs "name_list"

Example

$ srvctl unsetenv listener -envs "TNS_ADMIN"

---

srvctl unsetenv mgmtdb

Unsets the management database (CHM repository) environment configurations.

Syntax

srvctl unsetenv mgmtdb -envs "name_list"

Example

$ srvctl unsetenv mgmtdb -envs "LANG"

---

srvctl unsetenv mgmtlsnr

Unsets the management listener resource (for CHM) environment configurations.

Syntax

srvctl unsetenv mgmtlsnr -envs "name_list"

Use the -envs parameter to specify a comma-delimited list of the names of environment variables enclosed in double quotation marks ("").

vahidusefzadeh@gmail.com

Example

$ srvctl unsetenv mgmtlsnr -envs "LANG"

---

srvctl unsetenv nodeapps

Unsets the environment configuration for the node applications.

Syntax

srvctl unsetenv nodeapps -envs "name_list" [-viponly] [-gsdonly] [-onsonly] [-verbose]

Example

The following example unsets the environment configuration for all node applications:

$ srvctl unsetenv nodeapps -envs "test_var1,test_var2"

---

srvctl unsetenv vip

Unsets the environment configuration for the specified cluster VIP.

Syntax

srvctl unsetenv vip -vip "vip_name_list" -envs "name_list" [-verbose]

Example

The following example unsets the CLASSPATH environment variable for a cluster VIP:

$ srvctl unsetenv vip -vip "crm2-vip" -envs "CLASSPATH"

---

update

The srvctl update command requests that the specified running object use the new configuration information stored in the OCR.

---

vahidusefzadeh@gmail.com

srvctl update database

Updates the specified database to use the new listener endpoints.

Syntax

srvctl update database -db db_unique_name

Use the -db parameter to specify the unique name of the database to update.

---

srvctl update gns

Use the srvctl update gns command to modify a Grid Naming Service (GNS) instance.

Syntax

Use the srvctl update gns command with one of these syntax models:

srvctl update gns -advertise name -address ip_address [-timetolive time_to_live] [-verbose]

srvctl update gns -delete name [-address address] [-verbose]

srvctl update gns -alias alias -name name [-timetolive time_to_live] [-verbose]

srvctl update gns -deletealias alias [-verbose]

srvctl update gns -createsrv service -target target -protocol protocol [-weight weight] [-priority priority] [-port port_number] [-timetolive time_to_live] [-instance instance_name] [-verbose]

srvctl update gns -deletesrv service_name -target target -protocol protocol [-verbose]

srvctl update gns -createtxt name -target target [-timetolive time_to_live] [-namettl name_ttl] [-verbose]

srvctl update gns -deletetxt name -target target [-verbose]

srvctl update gns -createptr name -target target [-timetolive time_to_live] [-namettl name_ttl] [-verbose]

srvctl update gns -deleteptr name -target target [-verbose]

Example

The following command advertises a name with GNS:

srvctl update gns -advertise myname -address 192.168.1.45

---

vahidusefzadeh@gmail.com

srvctl update listener

Updates the listener to listen on the new endpoints.

Syntax

Use the srvctl update listener command with the following syntax:

---

srvctl update listener

This command does not accept any additional parameters, except for -help.

---

srvctl update scan_listener

Updates the SCAN listeners to listen on the new endpoints.

Syntax

Use the srvctl update scan_listener command with the following syntax:

```
srvctl update scan_listener
```

This command does not accept any additional parameters, except for -help.

---

upgrade

The upgrade command upgrades the resources types and resources from an older version to a newer version.

---

srvctl upgrade database

The srvctl upgrade database command upgrades the configuration of a database and all of its services to the version of the database home from where this command is run.

Syntax

```
srvctl upgrade database -db db_unique_name -oraclehome Oracle_home
```

vahidusefzadeh@gmail.com

--other statments in RAC

ALTER DATABASE DISABLE THREAD 2;

ALTER DATABASE ENABLE THREAD 2;

ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 GROUP 11 SIZE 100M, GROUP 12 SIZE 100M;

ALTER SYSTEM CHECKPOINT LOCAL;

ALTER SYSTEM CHECKPOINT GLOBAL;

ALTER SYSTEM ARCHIVE LOG CURRENT;

--KILL SESSION 'integer1, integer2[, @integer3]'

ALTER SYSTEM KILL SESSION '80, 4, @2';

alter system kill session 'sid, serial#, @inst' noreplay;

alter system disconnect session 'sid, serial#, @inst' noreplay

-- channel configuration

syntax

CONFIGURE DEVICE TYPE [disk | sbt] PARALLELISM number_of_channels;

example

CONFIGURE DEVICE TYPE sbt PARALLELISM 3;

CONFIGURE CHANNEL 1.. CONNECT 'dbauser/pwd@mydb_1';

CONFIGURE CHANNEL 2.. CONNECT 'dbauser/pwd@mydb_2';

CONFIGURE CHANNEL 3.. CONNECT 'dbauser/pwd@mydb_3';

OR

ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK CONNECT 'SYS/oracle@node1';

vahidusefzadeh@gmail.com

vahidusefzadeh@gmail.com

chapter 3

ASM

ASMCMD Instance Management

---

dsget

Retrieves the discovery diskstring value that is used by the Oracle ASM instance and its clients.

Syntax

dsget [--normal | --parameter | --profile [-f]]

Example

ASMCMD [+] > dsget

profile: /devices1/disk*

parameter: /devices2/disk*

---

dsset

Sets the discovery diskstring value that is used by the Oracle ASM instance and its clients. The specified diskstring must be valid for existing mounted disk groups. The updated value takes effect immediately.

Syntax

dsset [--normal | --parameter | --profile [-f]] diskstring

Example

ASMCMD [+] > dsset /devices1/disk*,/devices2/disk*

---

vahidusefzadeh@gmail.com

lsct

Lists information about current Oracle ASM clients from the V$ASM_CLIENT view. A client, such as Oracle Database or Oracle ASM Dynamic Volume Manager (Oracle ADVM), uses disk groups that are managed by the Oracle ASM instance to which ASMCMD is currently connected.

Syntax

lsct [--suppressheader] [-g] [diskgroup]

Example

ASMCMD [+] > lsct data

DB_Name Status Software_Version Compatible_version Instance_Name Disk_Group

+ASM CONNECTED 12.1.0.0.1 12.1.0.0.1 +ASM DATA

asmvol CONNECTED 12.1.0.0.1 12.1.0.0.1 +ASM DATA

usefdb CONNECTED 12.1.0.0.1 12.0.0.0.0 usefdb DATA

---

lsop

Lists the current operations on a disk group in an Oracle ASM instance.

Syntax

lsop displays information from the V$ASM_OPERATION view.

Example

ASMCMD [+] > lsop

Group_Name Dsk_Num State Power

DATA REBAL WAIT 2

---

lspwusr

vahidusefzadeh@gmail.com

List the users from the local Oracle ASM password file.

Syntax

lspwusr [--suppressheader]

Example

ASMCMD [+] > lspwusr

Username sysdba sysoper sysasm

SYS TRUE TRUE TRUE

ASMSNMP TRUE FALSE FALSE

---

orapwusr

Add, drop, or modify an Oracle ASM password file user.

Syntax

orapwusr { { { --add | --modify [--password] }[--privilege {sysasm|sysdba|sysoper}] } | --delete } user

Example

ASMCMD [+] > orapwusr --add --privilege sysdba hruser

---

pwcopy

Copies an Oracle ASM or database instance password file to the specified location.

Syntax

pwcopy {--asm |--dbuniquename string} source destination

Example

ASMCMD [+] > pwcopy --asm +DATA/orapwasm +FRA/orapwasm_bak

copying +DATA/orapwasm -> +FRA/orapwasm_bak

---

pwcreate

Creates an Oracle ASM or database instance password file at the specified location.

Syntax

pwcreate { --asm |--dbuniquename string } file_path sys_password

Example

ASMCMD [+] > pwcreate –-asm '+DATA/orapwasm' 'welcome'

---

pwdelete

Deletes an Oracle ASM or database instance password file.

Syntax

pwdelete { --asm |--dbuniquename string | file_path }

Example

ASMCMD [+] > pwdelete +FRA/orapwasm_bak

---

pwget

Returns the location of the password file for the Oracle ASM or database instance.

Syntax

pwget { --asm | --dbuniquename string }

Example

ASMCMD [+] > pwget --asm

+DATA/orapwasm

---

pwmove

Moves an Oracle ASM or database instance password file to the specified location.

Syntax

pwmove { --asm | --dbuniquename string } source destination

Example

ASMCMD [+] > pwmove --asm +FRA/orapwasm_bak +DATA/orapwasm

moving +FRA/orapwasm_bak -> +DATA/orapwasm

---

pwset

Sets the location of the password file for an Oracle ASM or database instance.

Syntax

pwset { --asm | --dbuniquename string } file_path

Example

ASMCMD [+] > pwset --asm +DATA/orapwasm

---

showclustermode

Displays the current mode of the Oracle ASM cluster.

Syntax

showclustermode

Example

ASMCMD [+] > showclustermode

ASM cluster : Flex mode disabled

---

showclusterstate

Displays the current state of the cluster.

Syntax

showclusterstate

Example

ASMCMD [+] > showclusterstate

---

showpatches

Lists the patches applied on the Oracle Grid Infrastructure home.

Syntax

showpatches

Example

ASMCMD [+] > showpatches

---

showversion

Displays the patch levels of the Oracle ASM cluster release and software.

Syntax

showversion [--releasepatch] [--softwarepatch]

Example

ASMCMD [+] > showversion --softwarepatch

ASM version : 12.1.0.1.0

Software patchlevel : 0

ASMCMD [+] > showversion --releasepatch

ASM version : 12.1.0.1.0

---

shutdown

Shuts down an Oracle ASM instance.

Syntax

shutdown [--normal | --abort|--immediate]

Example

ASMCMD [+] > shutdown --normal

ASMCMD [+] > shutdown --immediate

ASMCMD [+] > shutdown --abort

---

spbackup

Backs up an Oracle ASM SPFILE to a backup file.

Example

ASMCMD> spbackup +DATA/asm/asmparameterfile/registry.253.721810181 +DATA/spfileBackASM.bak

ASMCMD> spbackup +DATA/asm/asmparameterfile/registry.253.721810181 +FRA/spfileBackASM.bak

---

vahidusefzadeh@gmail.com

spcopy

Copies an Oracle ASM SPFILE from the source location to an SPFILE in the destination location.

Syntax

spcopy [-u] source destination

Example

ASMCMD> spcopy +DATA/asm/asmparameterfile/registry.253.721810181 +FRA/spfileCopyASM.ora

ASMCMD> spcopy +DATA/asm/asmparameterfile/registry.253.721810181
$ORACLE_HOME/dbs/spfileCopyASM.ora

ASMCMD> spcopy -u /oracle/product/11.2.0/grid/dbs/spfileTestASM.ora +DATA/ASM/spfileCopyASM.ora

---

spget

Retrieves the location of the Oracle ASM SPFILE from the Grid Plug and Play (GPnP) profile.

Example

ASMCMD [+] > spget

+DATA/ASM/ASMPARAMETERFILE/registry.253.813507611

---

spmove

Moves an Oracle ASM SPFILE from source to destination and automatically updates the GPnP profile.

Example

ASMCMD> spmove +DATA/spfileASM.ora /oracle/product/11.2.0/grid/dbs/spfileMoveASM.ora

ASMCMD> spmove /oracle/product/11.2.0/grid/dbs/spfile+ASM.ora +DATA/ASM/spfileMoveASM.ora

---

spset

vahidusefzadeh@gmail.com

Sets the location of the Oracle ASM SPFILE in the Grid Plug and Play (GPnP) profile.

Syntax

spset location

Example

ASMCMD> spset +DATA/asm/asmparameterfile/asmspfile.ora

---

startup

Starts up an Oracle ASM instance.

Syntax

startup [--nomount] [--restrict] [--pfile pfile]

Example

ASMCMD> startup --nomount --pfile asm_init.ora

ASMCMD File Management

---

cd

Changes the current directory to the specified directory.

Syntax

cd [dir]

Examples

ASMCMD [+data/hr] > cd +data/usefdb

ASMCMD [+data/usefdb] > cd DATAFILE

ASMCMD [+data/usefdb/DATAFILE] >cd ..

---

cp

Enables you to copy files between Oracle ASM disk groups and between a disk group and the operating system.

Syntax

cp src_file [rem_connect_str:]tgt_file

cp [rem_connect_str:]src_file tgt_file

Examples

ASMCMD [+] > cp +data/usefdb/datafile/EXAMPLE.265.691577295 /mybackups/example.bak

ASMCMD [+] > cp /mybackups/examples.bak +data/usefdb/datafile/myexamples.bak

ASMCMD [+] > cp +data/usefdb/datafile/EXAMPLE.265.691577295
sys@myserver.+ASM:/scratch/backup/myexamples.bak

Enter Password:

---

du

Displays the total space used for files in the specified directory and in the entire directory tree under the directory.

Syntax

du [--suppressheader] [dir]

Example

ASMCMD [+] > du data/usefdb

Used_MB Mirror_used_MB

1756 3519

vahidusefzadeh@gmail.com

---

find

Displays the absolute paths of all occurrences of the specified name pattern (with wildcards) in a specified directory and its subdirectories.

Syntax

find [--type type] dir pattern

Examples

ASMCMD [+] > find +data undo*

+data/USEFDB/DATAFILE/UNDOTBS1.258.691577151

ASMCMD [+] > find --type CONTROLFILE +data/usefdb *

+data/usefdb/CONTROLFILE/Current.260.691577263

---

ls

Lists the contents of an Oracle ASM directory, the attributes of the specified file, or the names and attributes of all disk groups.

Syntax

ls [--suppressheader] [-lsdtLg][--absolutepath][--reverse] [--permission][pattern]

Examples

ASMCMD [+] > ls +data/usefdb/datafile

ASMCMD [+] > ls -lt +data/usefdb/datafile

Type Redund Striped Time Sys Name

DATAFILE MIRROR COARSE JUL 13 08:00:00 Y EXAMPLE.265.691577295

ASMCMD [+] > ls -l +data/usefdb/datafile/sy*

```
Type      Redund  Striped  Time           Sys  Name

DATAFILE  MIRROR  COARSE   JUL 13 05:00:00  Y    SYSAUX.257.691577149

DATAFILE  MIRROR  COARSE   JUL 13 02:00:00  Y    SYSTEM.256.691577149
```

ASMCMD [+] > ls -s +data/usefdb/datafile

```
Block_Size  Blocks     Bytes      Space  Name

     8192   12801  104865792   214958080  EXAMPLE.265.691577295
```

ASMCMD [+] > ls --permission +data/usefdb/datafile

```
User  Group  Permission  Name

           rw-rw-rw-  EXAMPLE.265.691577295
```

---

lsof

Lists the open files of the local clients.

Syntax

lsof [--suppressheader] {-G diskgroup|--dbname db| -C instance}

Example

ASMCMD [+] > lsof -G data

```
DB_Name  Instance_Name  Path

usefdb     usefdb         +data/usefdb/controlfile/current.260.691577263
```

ASMCMD [+] > lsof -C +ASM

```
DB_Name  Instance_Name  Path

asmvol   +ASM          +data/VOLUME1.271.679226013
```

---

mkalias

Creates an alias for the specified system-generated file name.

Syntax

mkalias file alias

Example

ASMCMD [+data/usefdb/datafile] > mkalias SYSAUX.257.721811945 sysaux.f

ASMCMD [+data/usefdb/datafile] > ls --absolutepath

none => EXAMPLE.265.721812093

+DATA/USEFDB/DATAFILE/sysaux.f => SYSAUX.257.721811945

none => SYSTEM.256.721811945

none => UNDOTBS1.258.721811945

none => USERS.259.721811947

sysaux.f

---

mkdir

Creates Oracle ASM directories under the current directory.

Syntax

mkdir dir [dir . . .]

Example

ASMCMD [+data] > mkdir subdir1 subdir2

---

pwd

Displays the absolute path of the current directory.

Syntax

pwd

Example

ASMCMD [+data/usefdb/datafile] > pwd

+data/usefdb/datafile

---

rm

Deletes the specified Oracle ASM files and directories.

Syntax

rm [-f|-r] pattern [pattern...]

Example

ASMCMD [+data/usefdb/datafile] > rm myexamples.bak

ASMCMD [+data] > rm -r subdir2

You may delete multiple files and/or directories.

Are you sure? (y/n) y

---

rmalias

Removes the specified aliases, retaining the files that the aliases reference.

Syntax

rmalias [-r] alias [alias...]

Example

ASMCMD [+data/usefdb/datafile] > rmalias sysaux.f

ASMCMD Disk Group Management

---

chdg

Changes a disk group (adds disks, drops disks, resizes disks, or rebalances a disk group) based on an XML configuration file.

Syntax

chdg { config_file.xml | 'contents_of_xml_file' }

Example

```
<chdg name="data" power="3">
 <drop>
  <fg name="fg1"></fg>
  <dsk name="data_0001"/>
 </drop>
 <add>
  <fg name="fg2">
   <dsk string="/dev/disk5"/>
  </fg>
 </add>
</chdg>
```

ASMCMD [+] > chdg data_config.xml

ASMCMD [+] > chdg '<chdg name="data" power="3">

<drop><fg name="fg1"></fg><dsk name="data_0001"/></drop>

<add><fg name="fg2"><dsk string="/dev/disk5"/></fg></add></chdg>'

---

chkdg

Checks or repairs the metadata of a disk group.

Syntax
chkdg [--repair] diskgroup
Example

ASMCMD [+] > chkdg --repair data

---

dropdg

Drops a disk group.

Syntax

dropdg [-r [-f]] diskgroup
Example

ASMCMD [+] > dropdg -r -f data

ASMCMD [+] > dropdg -r fra

---

iostat

Displays I/O statistics for Oracle ASM disks in mounted disk groups.

Syntax

iostat [--suppressheader] [-et] [--io] [--region][-G diskgroup] [interval]
Example

ASMCMD [+] > iostat -G data

```
Group_Name  Dsk_Name   Reads      Writes

DATA        DATA_0000  180488192  473707520

DATA        DATA_0001  1089585152 469538816
```

ASMCMD [+] > iostat --io -G data

```
Group_Name  Dsk_Name   Reads  Writes

DATA        DATA_0000  2801   34918

DATA        DATA_0001  58301  35700
```

---

lsattr

Lists the attributes of a disk group.

Syntax

lsattr [--suppressheader][-G diskgroup] [-lm] [pattern]

Example

ASMCMD [+] > lsattr -lm -G data

```
Group_Name  Name                  Value      RO  Sys

DATA        access_control.enabled   FALSE      N   Y

DATA        access_control.umask     066        N   Y

DATA        au_size                  1048576    Y   Y

DATA        cell.smart_scan_capable  FALSE      N   N

DATA        compatible.advm          11.2.0.3.0  N   Y

DATA        compatible.asm           12.1.0.0.0  N   Y

DATA        compatible.rdbms         12.1.0.0.0  N   Y

DATA        content.check            FALSE      N   Y

DATA        content.type             data       N   Y
```

```
DATA      disk_repair_time       3.6h      N  Y
DATA      failgroup_repair_time  24.0h     N  Y
DATA      idp.boundary           auto      N  Y
DATA      idp.type               dynamic   N  Y
DATA      phys_meta_replicated   true      Y  Y
DATA      sector_size            512       Y  Y
DATA      thin_provisioned       FALSE     N  Y
```

ASMCMD [+] > lsattr -G fra -l %compat*

```
Name           Value
compatible.asm    12.1.0.0.0
compatible.rdbms  11.2.0.3.0
```

---

lsdg

Lists mounted disk groups and their information.

Syntax

lsdg [--suppressheader] [-g] [--discovery] [pattern]

Example

ASMCMD [+] > lsdg data

```
State    Type    Rebal  Sector  Block       AU  Total_MB  Free_MB  Req_mir_free_MB  Usable_file_MB
MOUNTED  NORMAL  N         512   4096  4194304    12288    8835          1117          3859
```

(continued)

```
Offline_disks  Voting_files  Name
            0          N  DATA
```

---

lsdsk

Lists Oracle ASM disks.

Syntax

lsdsk [--suppressheader] [-kptgMI] [-G diskgroup][--member|--candidate][--discovery][--statistics][pattern]

Example

```
ASMCMD [+] > lsdsk -t -G data

Create_Date  Mount_Date  Repair_Timer  Path

13-JUL-09    13-JUL-09   0          /devices/diska1

13-JUL-09    13-JUL-09   0          /devices/diska2

...

ASMCMD [+] > lsdsk -p -G data /devices/diska*

Group_Num  Disk_Num     Incarn  Mount_Stat  Header_Stat  Mode_Stat  State   Path

      1       0  2105454210  CACHED     MEMBER      ONLINE    NORMAL  /devices/diska1

      1       1  2105454199  CACHED     MEMBER      ONLINE    NORMAL  /devices/diska2

      1       2  2105454205  CACHED     MEMBER      ONLINE    NORMAL  /devices/diska3

ASMCMD [+] > lsdsk --candidate -p

Group_Num  Disk_Num     Incarn  Mount_Stat  Header_Stat  Mode_Stat  State   Path

      0       5  2105454171  CLOSED     CANDIDATE    ONLINE    NORMAL  /devices/diske1

      0      25  2105454191  CLOSED     CANDIDATE    ONLINE    NORMAL  /devices/diske2

...
```

---

vahidusefzadeh@gmail.com

lsod

Lists the open Oracle ASM disks.

Syntax

lsod [--suppressheader] [-G diskgroup] [--process process] [pattern]

Example

ASMCMD [+] > lsod -G data --process *LGWR*

Instance Process OSPID Path

1 oracle@myserver02 (LGWR) 26593 /devices/diska1

1 oracle@myserver02 (LGWR) 26593 /devices/diska2

ASMCMD [+] > lsod --process *LGWR* *diska*

Instance Process OSPID Path

1 oracle@myserver02 (LGWR) 26593 /devices/diska1

1 oracle@myserver02 (LGWR) 26593 /devices/diska2

---

md_backup

The md_backup command creates a backup file containing metadata for one or more disk groups.

Syntax

md_backup backup_file [-G 'diskgroup [,diskgroup,...]']

Example

ASMCMD [+] > md_backup /scratch/backup/alldgs20100422

Disk group metadata to be backed up: DATA

Disk group metadata to be backed up: FRA

Current alias directory path: USEFDB/ONLINELOG

Current alias directory path: USEFDB/PARAMETERFILE

Current alias directory path: USEFDB/ARCHIVELOG/2010_04_21

ASMCMD [+] > md_backup /scratch/backup/data20100422 -G data

Disk group metadata to be backed up: DATA

Current alias directory path: USEFDB/DATAFILE

Current alias directory path: USEFDB/TEMPFILE

---

md_restore

The md_restore command restores disk groups from a metadata backup file.

Syntax

md_restore backup_file [--silent] [--full|--nodg|--newdg -o 'old_diskgroup:new_diskgroup [,...]'] [-S sql_script_file] [-G 'diskgroup [,diskgroup...]']

Example

ASMCMD [+] > md_restore –-full –G data –-silent /scratch/backup/alldgs20100422

ASMCMD [+] > md_restore –-nodg –G data –-silent /scratch/backup/alldgs20100422

ASMCMD [+] > md_restore –-newdg -o 'data:data2' --silent /scratch/backup/data20100422

ASMCMD [+] > md_restore -S override.sql --silent /scratch/backup/data20100422

---

mkdg

Creates a disk group based on an XML configuration file.

Syntax

mkdg { config_file.xml | 'contents_of_xml_file' }

Example

<dg name="data" redundancy="normal">

```
 <fg name="fg1">
  <dsk string="/dev/disk1"/>
  <dsk string="/dev/disk2"/>
 </fg>
 <fg name="fg2">
  <dsk string="/dev/disk3"/>
  <dsk string="/dev/disk4"/>
 </fg>
 <a name="compatible.asm" value="11.2"/>
 <a name="compatible.rdbms" value="11.2"/>
 <a name="compatible.advm" value="11.2"/>
</dg>
```

```
ASMCMD [+] > mkdg data_config.xml
```

```
ASMCMD [+] > mkdg '<dg name="data"><dsk string="/dev/disk*"/></dg>'
```

---

mount

Mounts a disk group.

Syntax

```
mount [--restrict] { [-a] | [-f] diskgroup[ diskgroup ...] }
```

Example

```
ASMCMD [+] > mount -f data
```

```
ASMCMD [+] > mount --restrict data
```

ASMCMD [+] > mount -a

---

offline

Offline disks or failure groups that belong to a disk group.

Syntax

offline -G diskgroup { -F failgroup |-D disk} [-t {minutes | hours}]

Example

ASMCMD [+] > offline -G data -F failgroup1

ASMCMD [+] > offline -G data -D data_0001 -t 1.5h

---

online

Online all disks, a single disk, or a failure group that belongs to a disk group.

Syntax

online -G diskgroup { -a | -F failgroup |-D disk} [--power n] [-w]

Example

ASMCMD [+] > online -G data -F failgroup1 -w

ASMCMD [+] > online -G data -D data_0001

---

rebal

Rebalances a disk group.

Syntax

rebal [--power power] [-w] diskgroup

Example

ASMCMD [+] > rebal --power 4 fra

ASMCMD [+] > lsop

Group_Name Dsk_Num State Power

FRA   REBAL   RUN   4

---

remap

Marks a range of blocks as unusable on the disk and relocates any data allocated in that range.

Syntax

remap diskgroup disk block_range

Examples

ASMCMD [+] > remap DATA DATA_0001 5000-5999

ASMCMD [+] > remap FRA FRA_0002 6230-6339

---

setattr

Sets the attributes for an Oracle ASM disk group.

Syntax

setattr -G diskgroup attribute_name attribute_value

Example

ASMCMD [+] > setattr -G data compatible.asm 11.2.0.0.0

ASMCMD [+] > setattr -G data compatible.rdbms 11.2.0.0.0

---

umount

Dismounts a disk group.

Syntax

umount { -a | [-f] diskgroup }

Example

ASMCMD [+] > umount -a

ASMCMD [+] > umount -f data

ASMCMD File Access Control Commands

---

chgrp

Changes the user group of a file or list of files.

Syntax

chgrp usergroup file [file ...]

Examples

ASMCMD [+] > chgrp asm_data +data/usefdb/controlfile/Current.260.684924747

ASMCMD [+fra/usefdb/archivelog/flashback] > chgrp asm_fra log_7.264.684968167 log_8.265.684972027

---

chmod

Changes permissions of a file or list of files.

Syntax

chmod mode file [file ...]

Examples

ASMCMD [+fra/usefdb/archivelog/flashback] > chmod ug+rw log_7.264.684968167 log_8.265.684972027

ASMCMD [+fra/usefdb/archivelog/flashback] > chmod 640 log_7.264.684968167 log_8.265.684972027

ASMCMD [+] > ls --permission +fra/usefdb/archivelog/flashback

```
User     Group     Permission  Name
grid     asm_fra   rw-r-----  log_7.264.684968167
grid     asm_fra   rw-r-----  log_8.265.684972027
```

---

chown

Changes the owner of a file or list of files.

Syntax

chown user[:usergroup] file [file ...]

Examples

ASMCMD [+fra/usefdb/archivelog/flashback] > chown oracle1 log_7.264.684968167 log_8.265.684972027

ASMCMD [+fra/usefdb/archivelog/flashback] > chown oracle1:asm_fra log_9.264.687650269

---

groups

Lists all the user groups to which the specified user belongs.

Syntax

groups diskgroup user

Example

ASMCMD [+] > groups data oracle1

asm_data

---

grpmod

Adds or removes operating system (OS) users to and from an existing Oracle ASM user group.

Syntax

grpmod { --add | --delete } diskgroup usergroup user [user...]

Examples

ASMCMD [+] > grpmod –-add fra asm_fra oracle1 oracle2

ASMCMD [+] > grpmod –-delete data asm_data oracle2

---

lsgrp

Lists all Oracle ASM user groups or only groups that match a specified pattern.

Syntax

lsgrp [--suppressheader][-a] [-G diskgroup] [pattern]

Examples

ASMCMD [+] > lsgrp asm%

DG_Name Grp_Name Owner

FRA asm_fra grid

DATA asm_data grid

ASMCMD [+] > lsgrp -a

DG_Name Grp_Name Owner Members

FRA asm_fra grid oracle1

DATA asm_data grid oracle1 oracle2

vahidusefzadeh@gmail.com

---

lsusr

Lists Oracle ASM users in a disk group.

Syntax

lsusr [--suppressheader][-a] [-G diskgroup] [pattern]

Examples

ASMCMD [+] > lsusr -G data

User_Num OS_ID OS_Name

3 1001 grid

1 1021 oracle1

2 1022 oracle2

---

mkgrp

Creates a new Oracle ASM user group.

Syntax

mkgrp diskgroup usergroup [user] [user...]

Example

ASMCMD [+] > mkgrp data asm_data oracle1 oracle2

---

mkusr

Adds an operating system (OS) user to a disk group.

Syntax

mkusr diskgroup user

Examples

ASMCMD [+] > mkusr data oracle1

ASMCMD [+] > mkusr fra oracle2

---

passwd

Changes the password of a user.

Syntax

passwd user

Example

ASMCMD [+] > passwd oracle2

Enter old password (optional):

Enter new password: ******

---

rmgrp

Removes a user group from a disk group.

Syntax

rmgrp diskgroup usergroup

Example

ASMCMD [+] > rmgrp data asm_data

---

rmusr

Deletes an operating system (OS) user from a disk group.

Syntax

rmusr [-r] diskgroup user

Example

ASMCMD [+] > rmusr data oracle2

---

rpusr

Replaces one operating system (OS) user with another in a disk group.

Syntax

rpusr diskgroup user1 user2

Example

ASMCMD [+] > rpusr data oracle1 oracle2

ASMCMD Volume Management

---

volcreate

Creates an Oracle ADVM volume in the specified disk group.

Syntax

volcreate -G diskgroup -s size [--column number] [--width stripe_width] [--redundancy {high|mirror|unprotected}] [--primary {hot|cold}] [--secondary {hot|cold}] volume

Examples

ASMCMD [+] > volcreate -G data -s 10G --width 64K --column 8 volume1

ASMCMD [+] > volinfo -G data volume1

Diskgroup Name: DATA

Volume Name: VOLUME1

Volume Device: /dev/asm/volume1-123

State: ENABLED

Size (MB): 10240

Resize Unit (MB): 64

Redundancy: MIRROR

Stripe Columns: 8

Stripe Width (K): 64

Usage:

Mountpath:

---

voldelete

Deletes an Oracle ADVM volume.

Syntax

voldelete -G diskgroup volume

Example

ASMCMD [+] > voldelete -G data volume1

---

voldisable

Disables Oracle ADVM volumes in mounted disk groups and removes the volume device on the local node.

Syntax

voldisable { --all |{ -G diskgroup { -a | volume } } }

Example

ASMCMD [+] > voldisable -G data volume1

---

volenable

Enables Oracle ADVM volumes in mounted disk groups.

Syntax

volenable { --all |{ -G diskgroup { -a | volume } } }

Example

ASMCMD [+] > volenable -G data volume1

---

volinfo

Displays information about Oracle ADVM volumes.

Syntax

volinfo { --all |{ -G diskgroup { -a | volume } } }

volinfo {--show_diskgroup |--show_volume} volumedevice

Examples

ASMCMD [+] > volinfo -G data volume1

Diskgroup Name: DATA

Volume Name: VOLUME1

Volume Device: /dev/asm/volume1-123

State: ENABLED

Size (MB): 10240

Resize Unit (MB): 64

Redundancy: MIRROR

Stripe Columns: 8

Stripe Width (K): 64

Usage: ACFS

Mountpath: /acfsmounts/acfs1

ASMCMD [+] > volinfo -G data -a

Diskgroup Name: DATA

Volume Name: VOLUME1

Volume Device: \\.\asm-volume1-311

State: ENABLED

Size (MB): 1024

Resize Unit (MB): 32

Redundancy: MIRROR

Stripe Columns: 4

Stripe Width (K): 128

Usage: ACFS

Mountpath: C:\oracle\acfsmounts\acfs1

---

volresize

Resizes an Oracle ADVM volume.

Syntax

volresize -G diskgroup -s size [-f] volume

Example

ASMCMD [+] > volresize -G data -s 20G volume1

---

vahidusefzadeh@gmail.com

volset

Sets attributes of an Oracle ADVM volume in mounted disk groups.

Syntax

volset -G diskgroup [--usagestring string] [--mountpath mount_path] [--primary {hot|cold}] [--secondary {hot|cold}] volume

Example

ASMCMD [+] > volset -G data --usagestring 'no file system created' volume1

---

volstat

Reports I/O statistics for Oracle ADVM volumes.

Syntax

volstat [-G diskgroup] [volume]

Example

ASMCMD [+] > volstat -G data

DISKGROUP NUMBER / NAME: 1 / DATA

VOLUME_NAME

READS BYTES_READ READ_TIME READ_ERRS

WRITES BYTES_WRITTEN WRITE_TIME WRITE_ERRS

VOLUME1

10085 2290573312 22923 0

1382 5309440 1482 0

vahidusefzadeh@gmail.com

ACFS Commands

---

acfsload

acfsload loads or unloads Oracle ACFS, Oracle ADVM, and Oracle Kernel Services Driver (OKS) drivers.

Syntax

acfsload { start | stop } [-s]

Example

acfsload stop

---

acfsdriverstate

acfsdriverstate provides information on the current state of the Oracle ACFS, Oracle ADVM, and Oracle Kernel Services Driver (OKS) drivers.

Syntax

acfsdriverstate [-orahome ORACLE_HOME]{ installed | loaded | version | supported }

Example

$ acfsdriverstate -orahome /users/12.1.0/grid/ version

ACFS-9325: Driver OS kernel version = 2.6.18-8.el5xen(x86_64)

ACFS-9326: Driver Oracle version = 120209

---

acfsplugin

The acfsplugin application programming interface (API) sends and receives messages to and from the local plug-in enabled Oracle ACFS driver from the application plug-in module.

Syntax

sb8 acfsplugin_metrics(ub4 metric_type, ub1 *metrics, ub4 metric_buf_len, oratext *mountp);

Example

$ /sbin/acfsutil plugin enable -m acfsmetric1 -t HRDATA /humanresources

---

fsck

Checks and repairs an Oracle ACFS file system.

Syntax

fsck -t acfs -h /dev/null

fsck [-a|-f] [-v] [-vv] -t acfs [-n|-y] volume_device

Example

/sbin/fsck -a -v -y -t acfs /dev/asm/volume1-123

---

mkfs

Creates an Oracle ACFS file system.

Syntax

mkfs -t acfs -h

mkfs [-v] [-f]-t acfs [-n name] volume_device [size]

Example

ASMCMD [+] > volinfo -a

Volume Name: VOLUME1

Volume Device: /dev/asm/volume1-123

State: ENABLED

vahidusefzadeh@gmail.com

$ /sbin/mkfs -t acfs /dev/asm/volume1-123

---

mount

Mounts an Oracle ACFS file system.

Syntax

mount -h

mount [-v] -t acfs [-o options] volume_device mount_point

mount

Examples

/bin/mount -t acfs /dev/asm/volume1-123 /acfsmounts/acfs1

/bin/mount -t acfs -o all none none

---

umount

Dismounts an Oracle ACFS file system.

Syntax

umount -h
umount [-v] volume_device |mount_point
umount -a [-t acfs]
umount -h displays usage text and exits.

Examples

/bin/umount /dev/asm/volume1-123

/bin/umount /acfsmounts/acfs1

---

acfsutil tag info

Displays the tag names for tagged directories or file names in Oracle ACFS file systems.

vahidusefzadeh@gmail.com

Syntax

acfsutil tag info -h

acfsutil tag info [-r] [-c -t tagname] path [path ...]

acfsutil tag info [-c -t tagname]

Examples

$ /sbin/acfsutil tag info -r /acfsmounts/acfs1/myrepfiles/

---

acfsutil tag set

Adds the given tag to the specified files or directories in an Oracle ACFS file system

Syntax

acfsutil tag set -h
acfsutil tag set [-v] [-r] tagname path [path ...]

Example

$ /sbin/acfsutil tag set repl_grp1 -r /acfsmounts/acfs1/myrepfiles/*.dat

---

acfsutil tag unset

Removes the given tag name or all tag names from the specified file or directory.

Syntax

acfsutil tag unset -h
acfsutil tag unset [-v] [-r] {all | tagname} path [path ...]

Example

$ /sbin/acfsutil tag unset repl_grp1 -r

/acfsmounts/acfs1/myrepfiles/*.log

---

acfsutil repl bg

vahidusefzadeh@gmail.com

Starts, stops, or displays information about the Oracle ACFS replication background processes.

Syntax

acfsutil repl bg -h

acfsutil repl bg {start |stop|info} mount_point

Examples

$ /sbin/acfsutil repl bg start /acfsmounts/acfs1

$ /sbin/acfsutil repl bg stop /acfsmounts/acfs1

$ /sbin/acfsutil repl bg info /acfsmounts/acfs1

---

acfsutil repl compare

Verifies that files have been replicated on an Oracle ACFS file system.

Syntax

acfsutil repl compare -h

acfsutil repl compare [-a] [-t { all | tagname, ...}]primary_mount_point standby_mount_point

Example

$ /sbin/acfsutil repl compare /acfsmounts/acfs1 /nfs_mounted_standby

---

acfsutil repl info

Displays information about replication processing on an Oracle ACFS file system.

Syntax

acfsutil repl info -h

acfsutil repl info -c [-v] mount_point

acfsutil repl info -s [-v][-l] [-n number{m|h|d|w|y}] [-r start_time[#stop_time]] [-

f eventlog] mount_point

acfsutil repl info [-a|-e|-t] [-v][-l][-r start_time[#stop_time]] [-f eventlog] mount_point

Examples

$ /sbin/acfsutil repl info -c /acfsmounts/acfs1

$ /sbin/acfsutil repl info -s -n 5d /acfsmounts/acfs1

$ /sbin/acfsutil repl info -a -v /acfsmounts/acfs1

---

acfsutil repl init

Initiates replication on all the files in an Oracle ACFS file system or only those files with a specified list of tags.

Syntax

acfsutil repl init -h

acfsutil repl init primary -s standby_connect_string [tagname...] [-m standby_mount_point] [-c primary_service][-d trace_level] [-z {on|off}] mount_point

acfsutil repl init standby -p primary_connect_string[-c standby_service] [-d trace_level] mount_point

Examples

$ /sbin/acfsutil repl init primary

-s /@standby_repl_site

-m /standby/repl_data -c primary_repl_service

/acfsmounts/repl_data

$ /sbin/acfsutil repl init standby

-p /@primary_repl_site

-c standby_repl_service /standby/repl_data

---

acfsutil repl pause

Pauses replication on an Oracle ACFS file system.

Syntax

acfsutil repl pause -h

acfsutil repl pause mount_point

Examples

$ /sbin/acfsutil repl pause /acfsmounts/acfs1

---

acfsutil repl resume

Resumes replication on an Oracle ACFS file system where replication has been paused.

Syntax

acfsutil repl resume -h

acfsutil repl resume mount_point

Examples

$ /sbin/acfsutil repl resume /acfsmounts/acfs1

---

acfsutil repl sync

Synchronizes primary and standby file systems.

Syntax

acfsutil repl sync -h

acfsutil repl sync [apply] mount_point

Examples

$ /sbin/acfsutil repl sync /acfsmounts/acfs1

---

acfsutil repl terminate

Stops all replication activity on the Oracle ACFS file system at the site where it is run.

Syntax

acfsutil repl terminate -h

acfsutil repl terminate primary mount_point

acfsutil repl terminate standby [immediate] mount_point

Example

$ /sbin/acfsutil repl terminate /acfsmounts/acfs1

---

acfsutil repl trace

Sets the replication trace level for gathering trace information on an Oracle ACFS file system.

Syntax

acfsutil repl trace -h

acfsutil repl trace level mount_point

Example

$ /sbin/acfsutil repl trace 5 /acfsmounts/acfs1

---

acfsutil repl update

Updates replication information while replication is running on an Oracle ACFS file system.

Syntax

acfsutil repl update -h

acfsutil repl update [-p primary_connect_string] [-s standby_connect_string] [-z {on|off}] mount_point

Example

$ /sbin/acfsutil repl update -s mystandby@oracle.com /acfsmounts/acfs1

---

vahidusefzadeh@gmail.com

acfsutil audit archive

Forces an archival of the audit trail for the current host on the specified Oracle ACFS file system.

Syntax

acfsutil audit archive -h

acfsutil audit archive -m mount_point

Example

$ /sbin/acfsutil audit archive -m /acfsmounts/acfs1

---

acfsutil audit disable

Disables auditing for either Oracle ACFS encryption or security on a specified file system.

Syntax

acfsutil audit disable -h

acfsutil audit disable -m mount_point -s {encr |sec}

Example

$ /sbin/acfsutil audit disable -m /acfsmounts/acfs1 -s encr

---

acfsutil audit enable

Enables auditing for either Oracle ACFS encryption or security on a specified file system.

Syntax

acfsutil audit enable -h

acfsutil audit enable -m mount_point -s {encr |sec}

Example

$ /sbin/acfsutil audit enable -m /acfsmounts/acfs1 -s encr

---

vahidusefzadeh@gmail.com

acfsutil audit info

Displays auditing information.

Syntax

acfsutil audit info -h

acfsutil audit info [-m mount_point]

Example

$ /sbin/acfsutil audit info -m /acfsmounts/acfs1

Auditing information for '/acfsmounts/acfs1':

Audit trail size: 10MB

Archive File: READ

Audit Sources:

Security: ENABLED

Encryption: ENABLED

---

acfsutil audit init

Initializes Oracle ACFS auditing.

Syntax

acfsutil audit init -h

acfsutil audit init -M audit_manager_group -A auditor_group

Example

/sbin/acfsutil audit init -M myaudit_mgr_grp -A myauditor_grp

---

acfsutil audit purge

Purges the audit trail for a specified file system.

vahidusefzadeh@gmail.com

Syntax

acfsutil audit purge -h

acfsutil audit purge -m mount_point [-f]

Example

$ /sbin/acfsutil audit purge -m /acfsmounts/acfs1 -f

---

acfsutil audit read

Marks the audit trail to indicates to the audit manager that the log archive file for the current node has been reviewed, backed up as necessary, and is safe to purge.

Syntax

acfsutil audit read -h

acfsutil audit read -m mount_point

Example

$ /sbin/acfsutil audit read -m /acfsmounts/acfs1

---

acfsdbg

Debugs an Oracle ACFS file system.

Syntax

acfsdbg -h

acfsdbg [-r] [-l] volume_device

Example

$ /sbin/acfsdbg /dev/asm/voume1-123

acfsdbg: version = 11.2.0.3.0

Oracle ASM Cluster File System (ACFS) On-Disk Structure Version: 39.0

The ACFS volume was created at Mon Mar 2 14:57:45 2011

```
acfsdbg>

acfsbdg> calculate 60*1024

  61,440

  61440

  61440

  0xf000

  0170000

  1111:0000:0000:0000

acfsdbg> prompt "acfsdbg test>"

acfsdbg test>

echo "offset 64*1024" | acfsdbg /dev/asm/volume1-123
```

---

acfsutil info file

Displays information for a file in an Oracle ACFS file system.

Syntax

```
acfsutil info file -h
acfsutil info file [-d] [-o acfs_extent_offset] path
```

Example

```
$ /sbin/acfsutil info file /acfsmounts/acfs1/myfile
```

---

acfsutil info fs

Displays detailed Oracle ACFS file system information.

Syntax

vahidusefzadeh@gmail.com

acfsutil info fs -h

acfsutil info fs [{-o item|-s [interval [count]] }] [mount_point]

Example

$ /sbin/acfsutil info fs

/primary

ACFS Version: 11.2.0.2.0

flags: MountPoint,Available,Replication

mount time: Mon Oct 25 12:11:03 2010

volumes: 1

total size: 5368709120

total free: 4144230400

primary volume: /dev/asm/pvol-74

label:

flags: Primary,Available,ADVM

on-disk version: 40.0

allocation unit: 4096

major, minor: 252, 37889

size: 5368709120

free: 4144230400

ADVM diskgroup REPLDG

ADVM resize increment: 268435456

ADVM redundancy: unprotected

ADVM stripe columns: 4

ADVM stripe width: 131072

number of snapshots: 0

snapshot space usage: 0

replication status: primary

/standby

ACFS Version: 11.2.0.2.0

flags: MountPoint,Available,Replication

mount time: Mon Oct 25 12:11:03 2010

volumes: 1

total size: 5368709120

total free: 5263945728

primary volume: /dev/asm/svol-74

label:

flags: Primary,Available,ADVM

on-disk version: 40.0

allocation unit: 4096

major, minor: 252, 37890

size: 5368709120

free: 5263945728

ADVM diskgroup REPLDG

ADVM resize increment: 268435456

ADVM redundancy: unprotected

ADVM stripe columns: 4

ADVM stripe width: 131072

number of snapshots: 0

snapshot space usage: 0

replication status: standby

$ /sbin/acfsutil info fs -o mountpoints,replication

/primary

1

/standby

1

$ /sbin/acfsutil info fs -o mountpoints,isreplprimary

/primary

1

/standby

0

$ /sbin/acfsutil info fs -o mountpoints,isreplstandby

/primary

0

/standby

1

$ /sbin/acfsutil info fs -s /acfsmounts/acfs1

amount of change since mount: 359.22 MB

average rate of change since mount: 3 KB

$ /sbin/acfsutil info fs -s 60 4 /acfsmounts/acfs1

amount of change since mount: 359.22 MB

average rate of change since mount: 3 KB/s

amount of change: 15.02 MB rate of change: 256 KB/s

amount of change: 9.46 MB rate of change: 161 KB/s

amount of change: 7.32 MB rate of change: 125 KB/s

amount of change: 6.89 MB rate of change: 117 KB/s

...

vahidusefzadeh@gmail.com

---

acfsutil info id

Displays Oracle ACFS file system information for a given identifier and mount point.

Syntax

acfsutil info id -h

acfsutil info id num mount_point

Example

$ /sbin/acfsutil info id 117 /acfsmounts/acfs1

---

acfsutil plugin disable

Disables the Oracle ACFS plug-in infrastructure for an Oracle ACFS file system.

Syntax

acfsutil plugin disable -h

acfsutil plugin disable mount_point

Example

/sbin/acfsutil plugin disable /humanresource

---

acfsutil plugin enable

Enables the Oracle ACFS plug-in infrastructure for an Oracle ACFS file system.

Syntax

acfsutil plugin enable -h

acfsutil plugin enable -m metrictype [-t tag, ...] [-i interval[s|m]] mount_point

Example

/sbin/acfsutil plugin enable -m acfsmetric1 -t HRDATA /humanresource

vahidusefzadeh@gmail.com

---

acfsutil plugin info

Displays information about the Oracle ACFS plug-in infrastructure for an Oracle ACFS file system.

Syntax

acfsutil plugin info -h

acfsutil plugin info mount_point

Example

/sbin/acfsutil plugin info /humanresource

Plug-in status: ENABLED

Metric type: acfsmetric1

Enabled tags: HRDATA

Delivery method: Poll

Post interval (seconds):

/sbin/acfsutil plugin info /moviemods

Plug-in status: ENABLED

Metric type: acfsmetric1

Enabled tags: FILECONTENT

Delivery method: Post

Post interval (seconds): 120

---

acfsutil registry

Registers an Oracle ACFS file system with the Oracle ACFS mount registry.

Syntax

acfsutil registry -h

acfsutil registry

acfsutil registry -a [-c] [-f] [-t] [-u] [-n { nodes|all }]
[-o moptions] device mount_point
acfsutil registry -c [-t] [-u] [-o moptions] {device | mount_point}
acfsutil registry -d {device | mount_point}
acfsutil registry -l [device | mount_point]
acfsutil registry -m device

Example

$ /sbin/acfsutil registry -a /dev/asm/volume1-123 /acfsmounts/acfs1

$ /sbin/acfsutil registry -m /dev/asm/volume1-123

$ /sbin/acfsutil registry -d /dev/asm/volume1-123

---

acfsutil rmfs

Removes an Oracle ACFS file system.

Syntax

acfsutil rmfs -h
acfsutil rmfs device

Example

$ /sbin/acfsutil rmfs /dev/asm/volume1-123

---

acfsutil size

Resizes an Oracle ACFS file system.

Syntax

acfsutil size -h
acfsutil size [+|-]n[K|M|G|T|P] mount_point

Example

$ /sbin/acfsutil size +500M /acfsmounts/acfs1

vahidusefzadeh@gmail.com

---

acfsutil snap convert

Converts the type of an existing snapshot image from read-only to read-write or read-write to read-only.

Syntax

acfsutil snap convert -h

acfsutil snap convert [-r|-w] snapshot mount_point

Examples

$ acfsutil snap convert -w midday_test1 /acfsmounts/critical_apps

$ acfsutil snap convert -r midday_test2 /acfsmounts/critical_apps

---

acfsutil snap create

Creates a read-only or read-write snapshot of an Oracle ACFS file system or an existing snapshot.

Syntax

acfsutil snap create -h

acfsutil snap create [-r|-w] [-p parent_snapshot] snapshot mount_point

Examples

$ acfsutil snap create -w midday_test1 /acfsmounts/critical_apps

$ acfsutil snap create -w midday_test2 /acfsmounts/critical_apps

$ /sbin/acfsutil snap create payroll_report1 /acfsmounts/critical_apps

$ /sbin/acfsutil snap create payroll_report2 /acfsmounts/critical_apps

$ ls /acfsmounts/critical_apps/.ACFS/snaps

midday_test1 midday_test2 payroll_report1 payroll_report2

---

acfsutil snap delete

vahidusefzadeh@gmail.com

Deletes a snapshot of an Oracle ACFS file system.

Syntax

acfsutil snap delete -h

acfsutil snap delete snapshot mount_point

Example

$ /sbin/acfsutil snap delete midday_test1 /acfsmounts/critical_apps

acfsutil snap delete: Snapshot operation is complete.

---

acfsutil snap info

Displays information about Oracle ACFS file system snapshots.

Syntax

acfsutil snap info -h

acfsutil snap info -t [snapshot] mount_point

Example

$ /sbin/acfsutil snap info /acfsmounts/critical_apps

snapshot name: midday_test2

RO snapshot or RW snapshot: RW

parent name: /acfsmounts/critical_apps

snapshot creation time: Fri Feb 18 06:10:59 2011

snapshot name: payroll_report1

RO snapshot or RW snapshot: RO

parent name: /acfsmounts/critical_apps

snapshot creation time: Tue Feb 22 06:56:34 2011

snapshot name: payroll_report2

RO snapshot or RW snapshot: RO

parent name: /acfsmounts/critical_apps

snapshot creation time: Tue Feb 22 06:57:21 2011

number of snapshots: 3 (active)

1 (delete pending)

snapshot space usage: 8383348736

---

acfsutil tune

The acfsutil tune command displays the value of a specific tunable parameter or all Oracle ACFS tunable parameters, or sets the value of a tunable parameter in a persistent manner on a particular node.

Syntax

acfsutil tune -h

acfsutil tune [tunable_name]

acfsutil tune tunable_name=value

Examples

$ /sbin/acfsutil tune

$ acfsutil tune AcfsMaxOpenFiles=50000

---

advmutil canonical

advmutil canonical displays the canonical name of the specified Oracle ADVM device name.

Syntax

advmutil -h

advmutil canonical volume_device

Examples

[C:\]advmutil canonical asm-volume1-274

asm-volume1-274

vahidusefzadeh@gmail.com

```
[C:\]advmutil canonical \\.\asm-volume1-274

asm-volume1-274

[C:\]advmutil canonical \\?\asm-volume1-274

asm-volume1-274

[C:\]advmutil canonical \??\asm-volume1-274

asm-volume1-274
```

---

advmutil tune

advmutil tune displays the value of a specific Oracle ADVM parameter or sets the value of a specific Oracle ADVM parameter.

Syntax

advmutil -h

advmutil tune parameter [= value]

Examples

```
$ /sbin/advmutil tune deadlock_timer = 20

$ /sbin/advmutil tune deadlock_timer

  deadlock_timer = 20 (0x14)
```

---

advmutil volinfo

advmutil volinfo displays information about Oracle ADVM volume devices.

Syntax

advmutil -h

advmutil volinfo [-l][-L] [volume_device]

Examples

```
$ /sbin/advmutil volinfo /dev/asm/volume1-123
```

Device : /dev/asm/volume1-228

Interface Version: 1

Size (MB): 256

Resize Increment (MB): 32

Redundancy: mirror

Stripe Columns: 4

Stripe Width (KB): 128

Disk Group: DATA

Volume: VOLUME1

Compatible.advm : 11.2.0.0.0

$ /sbin/advmutil volinfo -l /dev/asm/volume1-228

Device : /dev/asm/volume1-228 : Interface Version : 1 : Size (MB) : 256 :

Resize Increment (MB) : 32 : Redundancy : mirror : Stripe Columns : 4 :

Stripe Width (KB) : 128 : Disk Group : DATA : Volume : VOLUME1 :

Compatible.advm : 11.2.0.0.0

$ /sbin/advmutil volinfo -L /dev/asm/volume1-228

/dev/asm/volume1-228 1 256 32 mirror 4 128 DATA VOLUME1 11.2.0.0.0

Creating Disk Groups

Syntax

CREATE DISKGROUP diskgroup_name

[{ HIGH | NORMAL | EXTERNAL } REDUNDANCY]

{ [QUORUM | REGULAR][FAILGROUP failgroup_name]

DISK qualified_disk_clause [, qualified_disk_clause]...

} ...

[ATTRIBUTE { 'attribute_name' = 'attribute_value' }...];

Example

```
CREATE DISKGROUP data NORMAL REDUNDANCY
  FAILGROUP controller1 DISK
    '/devices/diska1' NAME diska1
  FAILGROUP controller2 DISK
    '/devices/diskb1' NAME diskb1,
  ATTRIBUTE 'au_size'='4M',
    'compatible.asm' = '11.2',
    'compatible.rdbms' = '11.2',
    'compatible.advm' = '11.2';
```

OR

```
CREATE DISKGROUP ocr_data NORMAL REDUNDANCY
   FAILGROUP fg1 DISK '/devices/diskg1'
   QUORUM FAILGROUP fg3 DISK '/devices/diskg3'
   ATTRIBUTE 'compatible.asm' = '12.1.0.0.0';
```

Altering Disk Groups

ALTER DISKGROUP

```
ALTER DISKGROUP
  { { diskgroup_name
        { { add_disk_clause | drop_disk_clause }
```

```
          [, { add_disk_clause | drop_disk_clause } ]...
       | resize_disk_clause
      } [ rebalance_diskgroup_clause ]
  | { replace_disk_clause
  | rename_disk_clause
  | disk_online_clause
  | disk_offline_clause
  | rebalance_diskgroup_clause
  | check_diskgroup_clause
  | diskgroup_template_clauses
  | diskgroup_directory_clauses
  | diskgroup_alias_clauses
  | diskgroup_volume_clauses
  | diskgroup_attributes
  | modify_diskgroup_file
  | drop_diskgroup_file_clause
  | usergroup_clauses
  | user_clauses
  | file_permissions_clause
  | file_owner_clause
  | scrub_clause
  }
 | { diskgroup_name [, diskgroup_name ] ...
   | ALL
   } { undrop_disk_clause
```

```
    | diskgroup_availability
    | enable_disable_volume
    }
 }
```

Managing Oracle ADVM Volumes

```
SQL> ALTER DISKGROUP data ADD VOLUME volume1 SIZE 10G;

SQL> ALTER DISKGROUP data RESIZE VOLUME volume1 SIZE 15G;

SQL> ALTER DISKGROUP data DISABLE VOLUME volume1;

SQL> ALTER DISKGROUP data ENABLE VOLUME volume1;

SQL> ALTER DISKGROUP ALL DISABLE VOLUME ALL;

SQL> ALTER DISKGROUP data DROP VOLUME volume1;
```

Adding Disks to a Disk Group

```
ALTER DISKGROUP data1 ADD DISK '/devices/diska*';

OR

ALTER DISKGROUP data1 ADD DISK '/devices/diska5' NAME diska5,'/devices/diska6' NAME diska6,
'/devices/diska7' NAME diska7,'/devices/diska8' NAME diska8;

OR

ALTER DISKGROUP data1 ADD DISK '/devices/diskd*' REBALANCE POWER 5 WAIT;

OR

ALTER DISKGROUP data2 ADD DISK '/devices/diskc3' FORCE;
```

Replacing Disks in Disk Groups

```
SQL> ALTER DISKGROUP data2 REPLACE DISK diskc7 WITH '/devices/diskc18' POWER 3;
```

Renaming Disks in Disk Groups

SQL> ALTER DISKGROUP fra2 MOUNT RESTRICTED;

SQL> ALTER DISKGROUP fra2 RENAME DISK 'FRA1_0001' TO 'FRA2_0001', 'FRA1_0002' TO 'FRA2_0002';

ALTER DISKGROUP data1 DROP DISK diska5;

ALTER DISKGROUP data1 DROP DISK diska5 ADD FAILGROUP failgrp1 DISK '/devices/diska9' NAME diska9;

Intelligent Data Placement

ALTER DISKGROUP data ADD TEMPLATE datafile_hot ATTRIBUTE(HOT MIRRORHOT);

OR

ALTER DISKGROUP data MODIFY FILE '+data/usefdb/datafile/users.259.679156903' ATTRIBUTE(HOT MIRRORHOT);

Resizing Disks in Disk Groups

ALTER DISKGROUP data1 RESIZE DISKS IN FAILGROUP failgrp1 SIZE 100G;

Undropping Disks in Disk Groups

ALTER DISKGROUP data1 UNDROP DISKS;

Manually Rebalancing Disk Groups

ALTER DISKGROUP data2 REBALANCE POWER 5 WAIT;

Scrubbing Disk Groups

SQL> ALTER DISKGROUP data SCRUB POWER LOW;

SQL> ALTER DISKGROUP data SCRUB FILE '+DATA/USEFDB/DATAFILE/example.266.806582193' REPAIR POWER HIGH FORCE;

SQL> ALTER DISKGROUP data SCRUB DISK DATA_0005 REPAIR POWER HIGH FORCE;

Mounting and Dismounting Disk Groups

ALTER DISKGROUP ALL DISMOUNT;

ALTER DISKGROUP data1 MOUNT;

ALTER DISKGROUP data1 MOUNT FORCE;

exclusively mounts Disk Groups

ALTER DISKGROUP DATA MOUNT RESTRICTED;

vahidusefzadeh@gmail.com

online Disk Group

ALTER DISKGROUP DATA ONLINE DISK DATA_0000;

OFFLINE Disk Group

ALTER DISKGROUP DATA OFFLINE DISK DATA_0000 DROP AFTER 20 m;

rebalance operation

ALTER DISKGROUP GROUPC REBALANCE POWER 5;

Checking the Internal Consistency of Disk Group Metadata

ALTER DISKGROUP data1 CHECK ALL;

Dropping Disk Groups

DROP DISKGROUP diskgroup_name [FORCE INCLUDING CONTENTS | { INCLUDING | EXCLUDING } CONTENTS];

example

DROP DISKGROUP data1;

Drop diskgroup data01 force including contents;

Dropping Disk

alter diskgroup VOL1 drop disk 'VOL1_0000' force;

Renaming Disks Groups

renamedg dgname=fra1 newdgname=fra2 asm_diskstring='/devices/disk*' verbose=true

$ renamedg phase=one dgname=fra1 newdgname=fra2 asm_diskstring='/devices/disk*' config=/tmp/fra2.conf verbose=true

$ renamedg phase=two dgname=fra1 newdgname=fra2 config=/tmp/fra2.conf verbose=true

SQL> ALTER DISKGROUP fra2 RENAME DISKS ALL;

ALTER DISKGROUP with Compatibility Attributes

ALTER DISKGROUP data3 SET ATTRIBUTE 'compatible.asm' = '12.1';

Adding an Alias Name

ALTER DISKGROUP data ADD ALIAS '+data/usefdb/second.dbf' FOR
'+data/usefdb/datafile/mytable.342.123456789';

Renaming an Alias Name

ALTER DISKGROUP data RENAME ALIAS '+data/usefdb/datafile.dbf' TO '+data/payroll/compensation.dbf';

Dropping an Alias Name

ALTER DISKGROUP data DROP ALIAS '+data/payroll/compensation.dbf';

Dropping Files

ALTER DISKGROUP data DROP FILE '+data/payroll/compensation.dbf';

Creating a Directory

ALTER DISKGROUP data ADD DIRECTORY '+data/usefdb';

Renaming a Directory

ALTER DISKGROUP data RENAME DIRECTORY '+data/mydir' TO '+data/yourdir';

Dropping a Directory

ALTER DISKGROUP data DROP DIRECTORY '+data/yourdir' FORCE;

Checking the Internal Consistency of Disk Group Metadata

Alter diskgroup data01 check repair;

Alter diskgroup data01 check norepair;

ALTER DISKGROUP DATA CHECK;

vahidusefzadeh@gmail.com

chapter 4

EXPDP/IMPDP

ABORT_STEP

Used to stop the job after it is initialized. This allows the master table to be queried

before any data is exported.

Syntax

ABORT_STEP=[n | -1] ➔**Default**: Null

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=expdat.dmp SCHEMAS=hr **ABORT_STEP=-1**

impdp hr SCHEMAS=hr DIRECTORY=dpump_dir1 LOGFILE=schemas.log DUMPFILE=expdat.dmp **ABORT_STEP=-1**

ACCESS_METHOD

Instructs Export to use a particular method to unload data.

Syntax

ACCESS_METHOD=[AUTOMATIC | DIRECT_PATH | EXTERNAL_TABLE] ➔ **Default**: AUTOMATIC

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=expdat.dmp SCHEMAS=hr **ACCESS_METHOD=EXTERNAL_TABLE**

impdp hr SCHEMAS=hr DIRECTORY=dpump_dir1 LOGFILE=schemas.log DUMPFILE=expdat.dmp
ACCESS_METHOD=CONVENTIONAL

ATTACH

Attach to an existing job

Syntax

ATTACH [=[schema_name.]job_name] ➔**Default**: job currently in the user's schema, if there is only one

Example

vahidusefzadeh@gmail.com

expdp hr ATTACH= export_job

impdp hr ATTACH=import_job

CLUSTER

Determines whether Data Pump can use Oracle Real Application Clusters (Oracle

RAC) resources and start workers on other Oracle RAC instances.

Syntax

CLUSTER=[YES | NO] ➔ **Default**: YES

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_clus%U.dmp **CLUSTER=NO** PARALLEL=3

impdp hr DIRECTORY=dpump_dir1 SCHEMAS=hr **CLUSTER=NO** PARALLEL=3 NETWORK_LINK=dbs1

COMPRESSION

Reduce the size of a dumpfile.

Syntax

COMPRESSION=[ALL | DATA_ONLY | METADATA_ONLY | NONE] ➔**Default**: METADATA_ONLY

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_comp.dmp **COMPRESSION=METADATA_ONLY**

COMPRESSION_ALGORITHM

Specifies the compression algorithm to be used when compressing dump file data.

Syntax

COMPRESSION_ALGORITHM = {BASIC | LOW | MEDIUM | HIGH} ➔ **Default**: BASIC

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=usef.dmp COMPRESSION=DATA_ONLY

COMPRESSION_ALGORITHM=LOW

OR

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=usef.dmp COMPRESSION=ALL **COMPRESSION_ALGORITHM=BASIC**

CONTENT

Specifies data to unload.

Syntax

CONTENT=[ALL | DATA_ONLY | METADATA_ONLY]➔ **Default**: ALL

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=usef.dmp **CONTENT=METADATA_ONLY**

DATA_OPTIONS

The DATA_OPTIONS parameter designates how certain types of data should be handled

during export/import operations.

Syntax

DATA_OPTIONS=XML_CLOBS ➔ **Default**: There is no default.

Example

expdp hr TABLES=usef.xdb_tab1 DIRECTORY=dpump_dir1 DUMPFILE=hr_xml.dmp VERSION=11.2
DATA_OPTIONS=XML_CLOBS

impdp hr TABLES=employees CONTENT=DATA_ONLY DUMPFILE=dpump_dir1:table.dmp
DATA_OPTIONS=skip_constraint_errors

DIRECTORY

Directory object to be used for dumpfiles and logfiles.

Syntax

DIRECTORY=directory_object ➔ **Default**: DATA_PUMP_DIR

Example

expdp hr **DIRECTORY=dpump_dir1** DUMPFILE=employees.dmp CONTENT=METADATA_ONLY

impdp hr **DIRECTORY**=dpump_dir1 DUMPFILE=expfull.dmp LOGFILE=dpump_dir2:expfull.log

---

DUMPFILE

List of destination dump files

Syntax

DUMPFILE=[directory_object:]file_name [, ...] ➔ **Default**: expdat.dmp

Example

expdp hr SCHEMAS=hr DIRECTORY=dpump_dir1 **DUMPFILE**=dpump_dir2:exp1.dmp, exp2%U.dmp PARALLEL=3

impdp hr DIRECTORY=dpump_dir1 **DUMPFILE**=dpump_dir2:exp1.dmp, exp2%U.dmp

---

ENCRYPTION

Encrypt part or all of a dump file.

Syntax

ENCRYPTION = [ALL | DATA_ONLY | ENCRYPTED_COLUMNS_ONLY | METADATA_ONLY | NONE]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_enc.dmp JOB_NAME=enc1 **ENCRYPTION**=data_only **ENCRYPTION_PASSWORD**=usef

---

ENCRYPTION_ALGORITHM

Specify how encryption should be done.

Syntax

ENCRYPTION_ALGORITHM = [AES128 | AES192 | AES256] ➔ **Default**: AES128

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_enc3.dmp ENCRYPTION_PASSWORD=foobar **ENCRYPTION_ALGORITHM**=AES128

ENCRYPTION_MODE

vahidusefzadeh@gmail.com

Method of generating encryption key.

Syntax

ENCRYPTION_MODE = [DUAL | PASSWORD | TRANSPARENT]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_enc4.dmp ENCRYPTION=all ENCRYPTION_PASSWORD=secretwords ENCRYPTION_ALGORITHM=AES256 **ENCRYPTION_MODE**=DUAL

ENCRYPTION_PASSWORD

Password key for creating encrypted data within a dump file.

Syntax

ENCRYPTION_PASSWORD = password ➔ **Default**: There is no default

Example

expdp hr TABLES=employee_s_encrypt DIRECTORY=dpump_dir1 DUMPFILE=dpcd2be1.dmp ENCRYPTION=ENCRYPTED_COLUMNS_ONLY **ENCRYPTION_PASSWORD**=123456

impdp hr TABLES=employee_s_encrypt DIRECTORY=dpump_dir DUMPFILE=dpcd2be1.dmp **ENCRYPTION_PASSWORD**=123456

ENCRYPTION_PWD_PROMPT

Specifies whether Data Pump should prompt you for the encryption password.

Syntax

ENCRYPTION_PWD_PROMPT=[YES | NO] ➔**Default**: NO

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=usef.dmp **ENCRYPTION_PWD_PROMPT**=YES

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=usef.dmp **ENCRYPTION_PWD_PROMPT**=YES

ESTIMATE

Calculate job estimates.

Syntax

ESTIMATE=[BLOCKS | STATISTICS] ➔**Default**: BLOCKS

Example

expdp hr TABLES=employees **ESTIMATE**=STATISTICS DIRECTORY=dpump_dir1 DUMPFILE=estimate_stat.dmp

impdp hr TABLES=job_history NETWORK_LINK=source_database_link DIRECTORY=dpump_dir1 **ESTIMATE**=STATISTICS

ESTIMATE_ONLY

Calculate job estimates without performing the export.

Syntax

ESTIMATE_ONLY=[YES | NO] ➔**Default**: NO

Example

expdp hr **ESTIMATE_ONLY**=YES NOLOGFILE=YES SCHEMAS=HR

EXCLUDE

Exclude specific object types.

Syntax

EXCLUDE=object_type[:name_clause] [, ...]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_exclude.dmp **EXCLUDE**=VIEW,PACKAGE, FUNCTION

OR

EXCLUDE=FUNCTION

EXCLUDE=PROCEDURE

EXCLUDE=PACKAGE

EXCLUDE=INDEX:"LIKE 'EMP%' "

impdp system DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp PARFILE=exclude.par

---

FILESIZE

Specify the size of each dumpfile in units of bytes.

Syntax

FILESIZE=integer[B | KB | MB | GB | TB] ➔ **Default**: 0

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_3m.dmp **FILESIZE**=3MB

FLASHBACK_SCN

SCN used to reset session snapshot.

Syntax

FLASHBACK_SCN=scn_value

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_scn.dmp **FLASHBACK_SCN**=384632

impdp hr DIRECTORY=dpump_dir1 **FLASHBACK_SCN**=123456 NETWORK_LINK=source_database_link

---

FLASHBACK_TIME

Time used to find the closest corresponding SCN value.

Syntax

FLASHBACK_TIME="TO_TIMESTAMP(time-value)"

Example

DIRECTORY=dpump_dir1

DUMPFILE=hr_time.dmp

FLASHBACK_TIME="TO_TIMESTAMP('27-10-2012 13:16:00', 'DD-MM-YYYY HH24:MI:SS')"

impdp hr DIRECTORY=dpump_dir1 PARFILE=flashback_imp.par NETWORK_LINK=source_

database_link

vahidusefzadeh@gmail.com

FULL

Export/import entire database .

Syntax

FULL=[YES | NO]➔ **Default**: NO

Example

expdp hr DIRECTORY=dpump_dir2 DUMPFILE=expfull.dmp **FULL**=YES NOLOGFILE=YES

impdp hr DUMPFILE=dpump_dir1:expfull.dmp **FULL**=YES LOGFILE=dpump_dir2:full_imp.log

---

HELP

Displays online help for the Export/import utility.

Syntax

HELP = [YES | NO]

Example

expdp HELP = YES

impdp HELP = YES

INCLUDE

Include specific object types.

Syntax

INCLUDE = object_type[:name_clause] [, ...]

Example

SCHEMAS=HR

DUMPFILE=expinclude.dmp

DIRECTORY=dpump_dir1

LOGFILE=expinclude.log

INCLUDE=TABLE:"IN ('EMPLOYEES', 'DEPARTMENTS')"

INCLUDE=PROCEDURE

INCLUDE=INDEX:"LIKE 'EMP%'"

expdp hr PARFILE=usef.par

OR

expdp hr **INCLUDE**=TABLE DUMPFILE=dpump_dir1:exp_inc.dmp NOLOGFILE=YES

impdp system SCHEMAS=hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp PARFILE=imp_include.par

---

JOB_NAME

Name of export/import job

Syntax

JOB_NAME=jobname_string

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=exp_job.dmp **JOB_NAME**=exp_job NOLOGFILE=YES

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp **JOB_NAME**=impjob01

KEEP_MASTER

Indicates whether the master table should be deleted or retained at the end of a Data Pump job that completes successfully. The master table is automatically retained for jobs that do not complete successfully.

Syntax

KEEP_MASTER=[YES | NO] ➔**Default**: NO

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=expdat.dmp SCHEMAS=hr **KEEP_MASTER**=YES

impdp hr SCHEMAS=hr DIRECTORY=dpump_dir1 LOGFILE=schemas.log DUMPFILE=expdat.dmp **KEEP_MASTER**=YES

vahidusefzadeh@gmail.com

LOGFILE

Specifies the name, and optionally, a directory, for the log file of the export/import job.

Syntax

LOGFILE=[directory_object:]file_name

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=usef.dmp **LOGFILE**=hr_export.log

impdp hr SCHEMAS=HR DIRECTORY=dpump_dir2 **LOGFILE**=imp.log DUMPFILE=dpump_dir1:expfull.dmp

---

LOGTIME

Specifies that messages displayed during export/import operations be timestamped.

Syntax

LOGTIME=[NONE | STATUS | LOGFILE | ALL]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=expdat.dmp SCHEMAS=hr **LOGTIME**=ALL

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expdat.dmp SCHEMAS=hr **LOGTIME**=ALL
TABLE_EXISTS_ACTION=REPLACE

METRICS

Indicates whether additional information about the job should be reported to the Data

Pump log file.

Syntax

METRICS=[YES | NO]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=expdat.dmp SCHEMAS=hr **METRICS**=YES

impdp hr SCHEMAS=hr DIRECTORY=dpump_dir1 LOGFILE=schemas.log DUMPFILE=expdat.dmp **METRICS**=YES

NETWORK_LINK

Name of remote database link to the source system.

Syntax

NETWORK_LINK=source_database_link

Example

expdp hr DIRECTORY=dpump_dir1 **NETWORK_LINK**=source_database_link DUMPFILE=network_export.dmp LOGFILE=network_export.log

impdp hr TABLES=employees DIRECTORY=dpump_dir1 **NETWORK_LINK**=source_database_link EXCLUDE=CONSTRAINT

NOLOGFILE

Specifies whether to suppress creation of a log file.

Syntax

NOLOGFILE=[YES | NO] ➔ **Default**: NO

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=usef.dmp **NOLOGFILE**=YES

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp **NOLOGFILE**=YES

PARALLEL

Change the number of active workers for current job.

Syntax

PARALLEL=integer ➔ **Default**: 1

Example

expdp hr DIRECTORY=dpump_dir1 LOGFILE=parallel_export.log JOB_NAME=par4_job DUMPFILE=par_exp%u.dmp **PARALLEL**=4

impdp hr DIRECTORY=dpump_dir1 LOGFILE=parallel_import.log JOB_NAME=imp_par3 DUMPFILE=par_exp%U.dmp **PARALLEL**=3

vahidusefzadeh@gmail.com

PARFILE

Specify parameter file name.

Syntax

PARFILE=[directory_path]file_name ➔Default: There is no default

Example

SCHEMAS=HR

DUMPFILE=exp.dmp

DIRECTORY=dpump_dir1

LOGFILE=exp.log

expdp hr **PARFILE**=usef.par

impdp hr **PARFILE**=hr_imp.par

QUERY

Predicate clause used to export/import a subset of a table.

Syntax

QUERY = [schema.][table_name:] query_clause ➔Default: There is no default

Example

QUERY='sales:"WHERE EXISTS (SELECT cust_id FROM customers c WHERE cust_credit_limit > 10000 AND ku$.cust_id = c.cust_id)"'

OR

QUERY='sales:"WHERE EXISTS (SELECT cust_id FROM customers c WHERE cust_credit_limit > 10000 AND cust_id = c.cust_id)"'

expdp hr PARFILE=emp_query.par

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp PARFILE=query_imp.par NOLOGFILE=YES

REMAP_DATA

Specify a data conversion function.

Syntax

REMAP_DATA=[schema.]tablename.column_name:[schema.]pkg.function

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=remap1.dmp TABLES=employees
REMAP_DATA=usef.employees.employee_id:usef.remap.minus10
REMAP_DATA=usef.employees.first_name:usef.remap.plusx

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expschema.dmpTABLES=hr.employees
REMAP_DATA=hr.employees.first_name:hr.remap.plusx

REUSE_DUMPFILES

Overwrite destination dump file if it exists

Syntax

REUSE_DUMPFILES=[YES | NO]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=enc1.dmp TABLES=employees **REUSE_DUMPFILES**=YES

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp LOGFILE=reuse.log **REUSE_DATAFILES**=YES

REMAP_DATAFILE

Redefine datafile references in all DDL statements.

Syntax

REMAP_DATAFILE=source_datafile:target_datafile

Example

DIRECTORY=dpump_dir1

FULL=YES

DUMPFILE=db_full.dmp

REMAP_DATAFILE="'DB1$:[HRDATA.PAYROLL]tbs6.dbf':'/db1/hrdata/payroll/tbs6.dbf'"

impdp hr PARFILE=payroll.par

REMAP_SCHEMA

Objects from one schema are loaded into another schema.

Syntax

REMAP_SCHEMA=source_schema:target_schema

Example

impdp system DIRECTORY=dpump_dir1 DUMPFILE=hr.dmp **REMAP_SCHEMA**=hr:scott

---

REMAP_TABLE

Table names are remapped to another table.

Syntax

REMAP_TABLE=[schema.]old_tablename[.partition]:new_tablename

or

REMAP_TABLE=[schema.]old_tablename[:partition]:new_tablename

Example

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expschema.dmp TABLES=usef.employees
REMAP_TABLE=usef.employees:emps

REMAP_TABLESPACE

Tablespace object are remapped to another tablespace.

Syntax

REMAP_TABLESPACE=source_tablespace:target_tablespace

Example

impdp hr **REMAP_TABLESPACE**=tbs_1:tbs_6 DIRECTORY=dpump_dir1 DUMPFILE=employees.dmp

---

vahidusefzadeh@gmail.com

SQLFILE

Write all the SQL DDL to a specified file.

Syntax

SQLFILE=[directory_object:]file_name

Example

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp **SQLFILE**=dpump_dir2:expfull.sql

---

TABLE_EXISTS_ACTION

Action to take if imported object already exists.

Syntax

TABLE_EXISTS_ACTION=[SKIP | APPEND | TRUNCATE | REPLACE]

Example

impdp hr TABLES=employees DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp **TABLE_EXISTS_ACTION**=REPLACE

SAMPLE

Percentage of data to be exported.

Syntax

SAMPLE="HR"."EMPLOYEES":50

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=sample.dmp **SAMPLE**=70

SCHEMAS

List of schemas to export/import.

Syntax

SCHEMAS=schema_name [, ...]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=expdat.dmp **SCHEMAS**=hr,sh,oe

impdp hr **SCHEMAS**=hr DIRECTORY=dpump_dir1 LOGFILE=schemas.log DUMPFILE=expdat.dmp

---

SERVICE_NAME

Used to specify a service name to be used in conjunction with the CLUSTER parameter.

Syntax

SERVICE_NAME=name

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=hr_svname2.dmp **SERVICE_NAME**=sales

impdp system DIRECTORY=dpump_dir1 SCHEMAS=hr **SERVICE_NAME**=sales NETWORK_LINK=dbs1

SOURCE_EDITION

Edition to be used for extracting metadata

Syntax

SOURCE_EDITION=edition_name

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=exp_dat.dmp **SOURCE_EDITION**=exp_edition EXCLUDE=USER

impdp hr DIRECTORY=dpump_dir1 **SOURCE_EDITION**=exp_edition NETWORK_LINK=source_database_link EXCLUDE=USER

STATUS

Specifies the frequency at which the job status display is updated.

Syntax

STATUS=[integer]

Example

expdp hr DIRECTORY=dpump_dir1 SCHEMAS=hr,sh **STATUS**=300

impdp hr NOLOGFILE=YES **STATUS**=120 DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp

---

SKIP_UNUSABLE_INDEXES

Specifies whether Import skips loading tables that have indexes that were set to the

Index Unusable state.

Syntax

SKIP_UNUSABLE_INDEXES=[YES | NO]

Example

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp LOGFILE=skip.log **SKIP_UNUSABLE_INDEXES**=YES

TABLES

Specifies that you want to perform a table-mode export/import.

Syntax

TABLES=[schema_name.]table_name[:partition_name] [, ...]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=tables.dmp **TABLES**=employees,jobs,departments

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp **TABLES**=employees,jobs

---

TABLESPACES

Specifies a list of tablespace names to be exported/imported in tablespace mode.

Syntax

TABLESPACES=tablespace_name [, ...]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=tbs.dmp **TABLESPACES**=tbs_4, tbs_5, tbs_6

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp **TABLESPACES**=tbs_1,tbs_2,tbs_3,tbs_4

TRANSPORT_FULL_CHECK

Verify storage segments of all tables

Syntax

TRANSPORT_FULL_CHECK=[YES | NO]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=tts.dmp TRANSPORT_TABLESPACES=tbs_1 **TRANSPORT_FULL_CHECK**=YES LOGFILE=tts.log

impdp hr PARFILE=full_check.par

TRANSPORT_TABLESPACES

Specifies that you want to perform an export/import in transportable-tablespace mode.

Syntax

TRANSPORT_TABLESPACES=tablespace_name [, ...]

Example

expdp hr DIRECTORY=dpump_dir1 DUMPFILE=tts.dmp **TRANSPORT_TABLESPACES**=tbs_1

DIRECTORY=dpump_dir1

NETWORK_LINK=source_database_link

TRANSPORT_TABLESPACES=tbs_6

TRANSPORT_FULL_CHECK=NO

TRANSPORT_DATAFILES='user01/data/tbs6.dbf'

impdp hr PARFILE=tablespaces.par

TRANSPORTABLE

Specify whether transportable method can be used.

Syntax

TRANSPORTABLE = [ALWAYS | NEVER]

Example

expdp sh DIRECTORY=dpump_dir1 DUMPFILE=tto1.dmpTABLES=sh.sales2 **TRANSPORTABLE**=ALWAYS

impdp import_admin FULL=Y **TRANSPORTABLE**=ALWAYS VERSION=12 NETWORK_LINK=dbs1 ENCRYPTION_PASSWORD=password TRANSPORT_DATAFILES=<datafile_name> LOGFILE=dpump_dir1:fullnet.log

---

VERSION

Version of objects to export/import.

Syntax

VERSION=[COMPATIBLE | LATEST | version_string]

Example

expdp hr TABLES=usef.employees **VERSION**=LATEST DIRECTORY=dpump_dir1 DUMPFILE=emp.dmp NOLOGFILE=YES

impdp hr DIRECTORY=dpump_dir1 DUMPFILE=expfull.dmp TABLES=employees **VERSION**=LATEST

VIEWS_AS_TABLES

Specifies that one or more views are to be exported/imported as tables.

Syntax

VIEWS_AS_TABLES=[schema_name.]view_name[:table_name], ... ➔ **Default**: There is no default

Example

expdp scott/tiger **views_as_tables**=view1 directory=data_pump_dir dumpfile=scott1.dmp

impdp hr **VIEWS_AS_TABLES**=view1:view1_tab NETWORK_LINK=dblink1

vahidusefzadeh@gmail.com

vahidusefzadeh@gmail.com

chapter 5

RMAN

rman commands

```
rman
[ TARGET [=] ['] [userid][/[password]][@net_service_name] ']
| {CATALOG [=] ['] [userid][/[password]][@net_service_name] ']
| LOG [=] ']filename['] [APPEND]
...
]...
```

```
$ rman
$ rman NOCATALOG
$ rman TARGET SYS/pwd@target
$ rman TARGET SYS/pwd@target NOCATALOG
$ rman TARGET SYS/pwd@target LOG $ORACLE_HOME/dbs/my_log.log APPEND
$ rman CATALOG rman/pwd@catdb
$ rman TARGET=SYS/pwd@target CATALOG=rman/pwd@cat
$ rman TARGET / CATALOG rman/rman@cat
$ rman TARGET / SCRIPT dwh LOG /tmp/dwh.log
$ rman PIPE newpipe TARGET / TIMEOUT 90
$ rman @/my_dir/my_commands.txt
$ rman @backup_ts_generic.rman "/tmp" USERS
$ rman CMDFILE=backup_ts_users.rman
$ rman TARGET / @backup_db.rman
$ rman TARGET / CATALOG rman/pwd@cat CMDFILE cmdfile.rcv LOG outfile.txt
$ rman TARGET / CATALOG rman/pwd@cat DEBUG TRACE trace.log
$ rman TARGET SYS/pwd@prod CATALOG rman/rman@rcat @'/oracle/dbs/whole.rcv'
$ rman TARGET user/pwd CMDFILE=takefulldb.cmd @@takefulldb.cmd
$ rman CHECKSYNTAX @'/tmp/backup_db.cmd'
$ rman MSGNO
$ rman | tee rman.log
$ rman help=yes
```

@ (at sign) Run a command file.

@@ (double at sign) Run a command file in the same directory as another command file that is currently running. The @@ command differs from the @ command only when run from within a command file.

RMAN> @backup_db.rman

RMAN> @/my_dir/my_command_file.txt

RMAN> @/tmp/bkup_db.rman whole_db

RMAN> @backup_ts_generic.rman "/tmp" $1

RMAN> RUN {@backup_db.rman}

CONNECT command

Establish a connection between RMAN and a target, auxiliary, or recovery catalog database.

RMAN> CONNECT TARGET;

RMAN> CONNECT TARGET /

RMAN> CONNECT TARGET sys@tgt;

RMAN> CONNECT TARGET sys/pwd@tgt;

RMAN> CONNECT CATALOG rman@catdb;

RMAN> CONNECT CATALOG rman/pwd@catdb;

RMAN> CONNECT AUXILIARY /

RMAN> CONNECT AUXILIARY rman@auxdb;

RMAN> CONNECT AUXILIARY rman/pwd@auxdb;

CREATE CATALOG command

Create Oracle schema for the recovery catalog.

RMAN> CREATE CATALOG;

RMAN> CREATE CATALOG TABLESPACE rmants;

RMAN> CREATE VIRTUAL CATALOG;

SQL> EXEC rman.DBMS_RCVCAT.CREATE_VIRTUAL_CATALOG;

RMAN> SQL "EXEC catown.DBMS_RCVCAT.CREATE_VIRTUAL_CATALOG";

DROP CATALOG command

Remove Oracle schema from the recovery catalog.

RMAN> DROP CATALOG;

RESYNC CATALOG command

Perform a full resynchronization, which creates a snapshot control file and then copies any new or changed information from that snapshot control file to the recovery catalog.

RMAN> RESYNC CATALOG;

RMAN> RESYNC CATALOG FROM DB_UNIQUE_NAME prod_db;

RMAN> RESYNC CATALOG FROM DB_UNIQUE_NAME ALL;

UPGRADE CATALOG command

Upgrade the recovery catalog schema from an older version to the version required by the RMAN executable.

RMAN> UPGRADE CATALOG;

IMPORT CATALOG command

Import the metadata from one recovery catalog into another recovery catalog.

RMAN> IMPORT CATALOG cat@srcdb;

RMAN> IMPORT CATALOG rcat@inst DBID=2871507123;

RMAN> IMPORT CATALOG cat@srcdb DBID=1844750987, 61738563;

RMAN> IMPORT CATALOG cat@srcdb DB_NAME=prod2;

RMAN> IMPORT CATALOG cat@srcdb DB_NAME=prod3, prod4;

RMAN> IMPORT CATALOG rman/rman@catdb1 DB_NAME=prod1 NO UNREGISTER;

RMAN> IMPORT CATALOG rman/oracle@catdb1 NO UNREGISTER;

REGISTER command

Register the target database in the recovery catalog.

RMAN> REGISTER DATABASE;

RMAN> REGISTER CATALOG;

RMAN> REGISTER CATALOG TABLESPACE tbs-name;

---

vahidusefzadeh@gmail.com

UNREGISTER command

Unregister a Oracle database from the recovery catalog.

RMAN> UNREGISTER DATABASE;

RMAN> UNREGISTER DATABASE NOPROMPT;

RMAN> UNREGISTER DATABASE prod1;

RMAN> UNREGISTER DATABASE prod2 NOPROMPT;

RMAN> UNREGISTER DB_UNIQUE_NAME prod2;

RMAN> UNREGISTER DB_UNIQUE_NAME prod1 NOPROMPT;

RMAN> UNREGISTER DB_UNIQUE_NAME prod2 INCLUDING BACKUPS;

RMAN> UNREGISTER DB_UNIQUE_NAME prod3 INCLUDING BACKUPS NOPROMPT;

GRANT command

Grant privileges to a recovery catalog user.

RMAN> GRANT CATALOG FOR DATABASE prod1 TO vpc1;

RMAN> GRANT REGISTER DATABASE TO bckop2;

RMAN> GRANT RECOVERY_CATALOG_OWNER TO rmanop1, rmanop3;

REVOKE command

Revoke privileges from a recovery catalog user.

RMAN> REVOKE CATALOG FOR DATABASE prod1 FROM vpc1;

RMAN> REVOKE REGISTER DATABASE FROM bckop2;

RMAN> REVOKE RECOVERY_CATALOG_OWNER FROM bckop;

RESET DATABASE command

Inform RMAN that the SQL statement ALTER DATABASE OPEN RESETLOGS has been executed and that a new incarnation of the target database has been created, or reset the target database to a prior incarnation.

RMAN> RESET DATABASE TO INCARNATION 3;

STARTUP command

Startup the target database. This command is equivalent to the SQL*Plus STARTUP command.

RMAN> STARTUP;

RMAN> STARTUP PFILE=’/u01/app/oracle/admin/pfile/initsid.ora’

RMAN> STARTUP NOMOUNT;

RMAN> STARTUP MOUNT;

RMAN> STARTUP FORCE;

RMAN> STARTUP FORCE DBA;

RMAN> STARTUP FORCE DBA PFILE=c:\Oracle\Admin\pfile\init.ora;

RMAN> STARTUP FORCE NOMOUNT;

RMAN> STARTUP FORCE MOUNT DBA PFILE=/tmp/inittrgt.ora;

RMAN> STARTUP AUXILIARY nomount;

SHUTDOWN command

Shutdown the target database. This command is equivalent to the SQL*Plus SHUTDOWN command.

RMAN> SHUTDOWN;

RMAN> SHUTDOWN NORMAL;

RMAN> SHUTDOWN TRANSACTIONAL;

RMAN> SHUTDOWN IMMEDIATE;

RMAN> SHUTDOWN ABORT;

ALTER DATABASE command

Mount or open a database.

RMAN> ALTER DATABASE MOUNT;

RMAN> ALTER DATABASE OPEN;

RMAN> ALTER DATABASE OPEN RESETLOGS;

---

vahidusefzadeh@gmail.com

SHOW command

Display the current CONFIGURE settings.

```
SHOW
{ RETENTION POLICY
| BACKUP OPTIMIZATION
| [DEFAULT] DEVICE TYPE
| CONTROLFILE AUTOBACKUP [FORMAT]
| [AUXILIARY] CHANNEL [FOR DEVICE TYPE deviceSpecifier]
| MAXSETSIZE
| DATAFILE BACKUP COPIES
| ARCHIVELOG [BACKUP COPIES|DELETION POLICY]
| AUXNAME
| EXCLUDE
| ENCRYPTION {ALGORITHM | FOR [DATABASE|TABLESPACE]}
| COMPRESSION ALGORITHM
| SNAPSHOT CONTROLFILE NAME
| DB_UNIQUE_NAME
| ALL
} FOR [DB_UNIQUE_NAME ['db_unique_name'|ALL]];
```

```
RMAN> SHOW ALL;
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default
CONFIGURE BACKUP OPTIMIZATION OFF; # default
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default
CONFIGURE CONTROLFILE AUTOBACKUP OFF; # default
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO '%F'; # default
CONFIGURE DEVICE TYPE DISK PARALLELISM 1 BACKUP TYPE TO BACKUPSET; # default
CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE DISK TO 1; # default
CONFIGURE ARCHIVELOG BACKUP COPIES FOR DISK TO 1; # default
CONFIGURE MAXSETSIZE TO UNLIMITED; # default
```

CONFIGURE DEVICE TYPE DISK PARALLELISM 1; # default

CONFIGURE DATAFILE BACKUP COPIES FOR SBT TO 1; # default

CONFIGURE ARCHIVELOG BACKUP COPIES FOR SBT TO 1; # default

CONFIGURE ENCRYPTION FOR DATABASE OFF; # default

CONFIGURE ENCRYPTION ALGORITHM 'AES128'; # default

CONFIGURE COMPRESSION ALGORITHM 'BASIC' AS OF RELEASE 'DEFAULT' OPTIMIZE FOR LOAD TRUE; # default

CONFIGURE ARCHIVELOG DELETION POLICY TO NONE; # default

CONFIGURE SNAPSHOT CONTROLFILE NAME TO '.../dbs/snapcf_sid.f'; # default

%F = dbid, day, month, year and sequence

%U = %u_%p_%c

%u = eight characters of the backup set and time ...

%p = piece number within the backupset

%c = copy number of the backup piece ...

RMAN> SHOW RETENTION POLICY;

RMAN> SHOW RETENTION POLICY FOR DB_UNIQUE_NAME ALL;

RMAN> SHOW DEVICE TYPE;

RMAN> SHOW DEVICE TYPE FOR DB_UNIQUE_NAME prod3;

RMAN> SHOW DEFAULT DEVICE TYPE;

RMAN> SHOW CHANNEL;

RMAN> SHOW MAXSETSIZE;

RMAN> SHOW BACKUP OPTIMIZATION;

RMAN> SHOW SNAPSHOT CONTROLFILE NAME;

RMAN> SHOW CONTROLFILE AUTOBACKUP;

RMAN> SHOW COMPRESSION ALGORITHM;

RMAN> SHOW ENCRYPTION ALGORITHM;

RMAN> SHOW ALL FOR DB_UNIQUE_NAME ALL;

RMAN> SHOW ALL FOR DB_UNIQUE_NAME 'STANDBY';

---

vahidusefzadeh@gmail.com

CONFIGURE command

To configure persistent RMAN settings. These settings apply to all RMAN sessions until explicitly changed or disabled.

```
CONFIGURE deviceConf;
CONFIGURE backupConf;
CONFIGURE AUXNAME FOR DATAFILE datafileSpec {TO 'filename' | CLEAR};
CONFIGURE SNAPSHOT CONTROLFILE NAME {TO 'filename' | CLEAR};
CONFIGURE cfauConf;
CONFIGURE ARCHIVELOG DELETION POLICY
  {CLEAR | TO {APPLIED ON [ALL] STANDBY | BACKED UP integer TIMES TO DEVICE TYPE deviceSpecifier
| NONE | SHIPPED TO [ALL] STANDBY}
        [{APPLIED ON [ALL] STANDBY | BACKED UP integer TIMES TO DEVICE TYPE deviceSpecifier |
NONE | SHIPPED TO [ALL] STANDBY}] …
  }

deviceConf::=
{ DEFAULT DEVICE TYPE { TO deviceSpec | CLEAR }
| DEVICE TYPE deviceSpec { PARALLELISM integer | CLEAR }
| [AUXILIARY] CHANNEL [integer] DEVICE TYPE deviceSpec {allocOperandList|CLEAR}
}

allocOperandList::=
{ PARMS [=] 'channel_parms'
| FORMAT [=] 'format_string' [, 'format_string']...
| { MAXPIECESIZE [=] integer | RATE [=] integer } [K | M | G]
...
}...

connectStringSpec::=
['] [userid] [/[password]] [@net_service_name] ['] 
```

vahidusefzadeh@gmail.com

backupConf::=

{RETENTION POLICY {TO {RECOVERY WINDOW OF integer DAYS

| REDUNDANCY [=] integer | NONE

}

| CLEAR

}

| MAXSETSIZE {TO {integer [K | M | G]| UNLIMITED}

| CLEAR

}

| {ARCHIVELOG | DATAFILE}

BACKUP COPIES FOR DEVICE TYPE deviceSpec {TO integer | CLEAR}

| BACKUP OPTIMIZATION {ON | OFF | CLEAR}

| EXCLUDE FOR TABLESPACE tablespace_name [CLEAR]

}

cfauConf::==

CONTROLFILE AUTOBACKUP {ON | OFF | CLEAR | FORMAT FOR DEVICE TYPE deviceSpec {TO 'format string'|CLEAR}}

```
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP OFF;
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO 'cf%F';
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO '+BACKUP';
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK CLEAR;
RMAN> CONFIGURE RETENTION POLICY TO REDUNDANCY 3;
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 7 DAYS;
RMAN> CONFIGURE RETENTION POLICY CLEAR;
RMAN> CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE DISK TO 2;

RMAN> CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE DISK TO 2;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY CLEAR;    --11g
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO NONE;
```

www.usefzadeh.com

```
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO SHIPPED TO STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO SHIPPED TO ALL STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON ALL STANDBY;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO BACKED UP 2 TIMES TO sbt;
RMAN> CONFIGURE ARCHIVELOG DELETION POLICY TO BACKED UP 3 TIMES TO disk;

RMAN> CONFIGURE DEFAULT DEVICE TYPE TO sbt;
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK;
RMAN> CONFIGURE DEVICE TYPE sbt PARALLELISM 3;
RMAN> CONFIGURE DEVICE TYPE DISK PARALLELISM 4;
RMAN> CONFIGURE DEVICE TYPE DISK PARALLELISM 3 BACKUP TYPE TO BACKUPSET;
RMAN> CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO COMPRESSED BACKUPSET;

RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt;
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt PARMS='ENV=mml_env_settings';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt PARMS 'ENV=(NSR_SERVER=bksrv1)';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt PARMS 'BLKSIZE=1048576';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt FORMAT 'bkup_%U';
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt CLEAR;
RMAN> CONFIGURE CHANNEL 2 DEVICE TYPE sbt CONNECT 'SYS/pwd@node2' PARMS
'ENV=(NSR_SERVER=bksrv2)';
RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT '/tmp/%U';
RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT 'C:\backup\df%t_s%s_s%p';
RMAN> CONFIGURE CHANNEL 2 DEVICE TYPE DISK FORMAT '/backup/db_%s%d_%p';
RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK FORMAT CLEAR;
RMAN> CONFIGURE CHANNEL DEVICE TYPE DISK DEBUG 5;

RMAN> CONFIGURE BACKUP OPTIMIZATION ON;
RMAN> CONFIGURE BACKUP OPTIMIZATION OFF;
RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO ‘/backup/snapcf_%d.f‘;
RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO ‘+FRA/snap/snapcf_%d.f‘;
```

RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO ‘/ocfs/oradata/snapcf‘;

RMAN> CONFIGURE SNAPSHOT CONTROLFILE NAME TO ‘/dev/sda‘;

RMAN> CONFIGURE MAXSETSIZE TO 100M;

RMAN> CONFIGURE MAXSETSIZE TO UNLIMITED;

RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt MAXPIECESIZE 1G;

RMAN> CONFIGURE EXCLUDE FOR TABLESPACE example;

RMAN> CONFIGURE EXCLUDE CLEAR;

RMAN> CONFIGURE AUXNAME FOR DATAFILE 4 TO '/oracle/auxfiles/aux_4.f';

RMAN> CONFIGURE AUXNAME FOR DATAFILE 2 CLEAR;

RMAN> CONFIGURE COMPRESSION ALGORITHM 'BZIP2';

RMAN> CONFIGURE COMPRESSION ALGORITHM 'ZLIB';

RMAN> CONFIGURE COMPRESSION ALGORITHM 'LOW'; --11g R2,corresponds to LZO

RMAN> CONFIGURE COMPRESSION ALGORITHM 'MEDIUM'; --11g R2,corresponds to ZLIB

RMAN> CONFIGURE COMPRESSION ALGORITHM 'HIGH'; --11g R2,corresponds to unmodified BZIP2

RMAN> CONFIGURE COMPRESSION ALGORITHM 'BASIC'; --Oracle 11g R2,corresponds to BZIP2

RMAN> CONFIGURE DB_UNIQUE_NAME 'standby' CONNECT IDENTIFIER 'standby_cs';

RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK FOR DB_UNIQUE_NAME 'standby';

RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK FOR DB_UNIQUE_NAME ALL;

RMAN> CONFIGURE DEFAULT DEVICE TYPE TO SBT FOR DB_UNIQUE_NAME po;

SET command

Set the value of various attributes that affect RMAN behaviour for the duration of a RUN block or a session.

SET {set_rman_option [;] | set_run_option;}

set_rman_option::=

{ECHO {ON|OFF} | DBID [=] integer

| CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE deviceSpec TO 'frmt_string'

```
set_run_option::=
{ NEWNAME FOR DATAFILE datafileSpec TO {'filename' | NEW}
| ARCHIVELOG DESTINATION TO 'log_archive_dest'
| untilClause
| COMMAND ID TO 'string'
| CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE deviceSpec TO 'frmt_string'
...
}
```

ECHO - Controls whether RMAN commands are displayed in the message log.

DBID - A unique 32-bit identification number computed when the database is created. RMAN displays the DBID upon connection to the target database. We can obtain the DBID by querying V$DATABASE or RC_DATABASE.

NEWNAME FOR DATAFILE - The default name for all subsequent RESTORE or SWITCH commands that affect the specified datafile.

MAXCORRUPT FOR DATAFILE - A limit on the number of previously undetected physical block corruptions that Oracle will allow in the datafile(s).

AUTOLOCATE - Force RMAN to automatically discover which nodes of an Oracle Real Application Clusters configuration contain the backups that you want to restore.

```
RMAN> SET ECHO ON;
RMAN> SET ECHO OFF;
RMAN> SET DATABASE prod;
RMAN> SET DBID=4240978820;
RMAN> SET DBID 591329635;
RMAN> SET COMMAND ID TO 'rman';
RMAN> SET MAXCORRUPT FOR DATABASE TO 2;
```

vahidusefzadeh@gmail.com

```
RMAN> SET MAXCORRUPT FOR DATAFILE 13 TO 200;
RMAN> SET BACKUP COPIES = 2;

RMAN> SET NEWNAME FOR DATABASE TO '/oradata1/%b';
RMAN> SET NEWNAME FOR TABLESPACE users TO '/oradata2/%U';
RMAN> SET NEWNAME FOR DATAFILE 1 to ‘/oradata/system01.dbf’;
RMAN> SET NEWNAME FOR DATAFILE '/disk7/tbs11.f' TO '/disk9/tbs11.f';
RMAN> SET NEWNAME FOR TEMPFILE 1 TO '/newdisk/dbs/temp1.f';

RMAN> SET CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE sbt TO 'cf_%F';
RMAN> SET CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO 'cf_%F.bak';
RMAN> SET UNTIL TIME ’04-23-2010:23:50:04’;
RMAN> SET ARCHIVELOG DESTINATION TO '/oracle/temp_restore';
RMAN> SET COMPRESSION ALGORITHM 'LOW';
RMAN> SET COMPRESSION ALGORITHM 'LOW' OPTIMIZE FOR LOAD FALSE;
RMAN> SET COMPRESSION ALGORITHM 'MEDIUM';
RMAN> SET COMPRESSION ALGORITHM 'HIGH';
```

BACKUP command

Backs up Oracle database files, copies of database files, archived logs, or backup sets.

```
BACKUP FULL Options
BACKUP FULL AS (COPY | BACKUPSET) Options
BACKUP INCREMENTAL LEVEL [=] integer Options
BACKUP INCREMENTAL LEVEL [=] integer AS (COPY | BACKUPSET) Options
BACKUP AS (COPY | BACKUPSET) Options
BACKUP AS (COPY | BACKUPSET) (FULL | INCREMENTAL LEVEL [=] integer) Options

Options::=
[backupOperand [backupOperand]...] backupSpec [backupSpec]...
[PLUS ARCHIVELOG [backupSpecOperand [backupSpecOperand]...]];
```

vahidusefzadeh@gmail.com

backupOperand::=

{ FORMAT [=] 'format_string' [, 'format_string']...

| CHANNEL ['] channel_id ']

| CUMULATIVE

| MAXSETSIZE [=] integer [K | M | G]

| TAG [=] ['] tag_name ']

| keepOption

| SKIP {OFFLINE | READONLY | INACCESSIBLE}

| VALIDATE

| NOT BACKED UP [SINCE TIME [=] 'date_string']

| COPIES [=] integer

| DEVICE TYPE deviceSpecifier

...

}

backupSpec::=

[(]

{ BACKUPSET

{ {ALL | completedTimeSpec } | primary_key) [, primary_key]... }

| COPY OF { DATABASE

| TABLESPACE ['] tablespace_name ['] [, ['] tablespace_name ']]...

| DATAFILE datafileSpec [, datafileSpec]...

}

| DATAFILE datafileSpec [, datafileSpec]...

| DATAFILECOPY 'filename' [, 'filename']...

| DATAFILECOPY FROM TAG [=] ['] tag_name ['] [, ['] tag_name ']]...

| DATAFILECOPY { ALL | LIKE 'string_pattern' }

| TABLESPACE ['] tablespace_name ['] [, ['] tablespace_name ']]...

| DATABASE

| archivelogRecordSpecifier

| CURRENT CONTROLFILE [FOR STANDBY]

```
| CONTROLFILECOPY 'filename'
| SPFILE
}
[backupSpecOperand [backupSpecOperand]...]

backupSpecOperand::=
{ FORMAT [=] 'format_string' [, 'format_string']...
| CHANNEL ['] channel_id ']
| CUMULATIVE
| MAXSETSIZE [=] integer [K | M | G]
| TAG [=] ['] tag_name ']
| keepOption
| SKIP {OFFLINE | READONLY | INACCESSIBLE}
| NOT BACKED UP [SINCE TIME [=] 'date_string' | integer TIMES]
| DELETE [ALL] INPUT
...
}

RMAN> BACKUP DATABASE;
RMAN> BACKUP DATABASE TAG=’test backup’;
RMAN> BACKUP DATABASE COMMENT=’full backup’;
RMAN> BACKUP TAG 'weekly_full_db_bkup' DATABASE MAXSETSIZE 10M;
RMAN> BACKUP MAXSETSIZE 500M DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DURATION 00:60 DATABASE;
RMAN> BACKUP DURATION 00:30 MINIMIZE TIME DATABASE;
RMAN> BACKUP DURATION 00:45 MINIMIZE LOAD DATABASE;

RMAN> BACKUP DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DATABASE KEEP FOREVER;
RMAN> BACKUP DATABASE KEEP UNTIL TIME=’SYSDATE+30’;
RMAN> BACKUP DATABASE UNTIL 'SYSDATE+365' NOLOGS;
RMAN> BACKUP DATABASE NOEXCLUDE;
```

```
RMAN> BACKUP DATABASE NOEXCLUDE KEEP FOREVER TAG=’abc’;
RMAN> BACKUP DATABASE SKIP READONLY;
RMAN> BACKUP DATABASE SKIP OFFLINE;
RMAN> BACKUP DATABASE SKIP INACCESSIBLE;
RMAN> BACKUP DATABASE SKIP READONLY SKIP OFFLINE SKIP INACCESSIBLE;
RMAN> BACKUP DATABASE FORCE; -- backup read only database also
RMAN> BACKUP DATABASE NOT BACKED UP;
RMAN> BACKUP DATABASE NOT BACKED UP SINCE TIME=’SYSDATE–3’;
RMAN> BACKUP NOT BACKED UP SINCE TIME 'SYSDATE-10' MAXSETSIZE 500M DATABASE PLUS
ARCHIVELOG;

RMAN> BACKUP DATABASE COPIES=2;
RMAN> BACKUP DATABASE FORMAT '/disk1/backups/db_%U.bck'
TAG quarterly KEEP UNTIL TIME 'SYSDATE+365' RESTORE POINT Q1FY12;
RMAN> BACKUP DEVICE TYPE DISK DATABASE;
RMAN> BACKUP DEVICE TYPE sbt DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DEVICE TYPE sbt DATAFILECOPY FROM TAG 'latest' FORMAT 'df%f_%d';
RMAN> BACKUP DEVICE TYPE sbt ARCHIVELOG LIKE '/disk%arc%' DELETE ALL INPUT;
RMAN> BACKUP DEVICE TYPE sbt BACKUPSET COMPLETED BEFORE 'SYSDATE-14'DELETE INPUT;
RMAN> BACKUP CHECK LOGICAL DATABASE;
RMAN> BACKUP VALIDATE CHECK LOGICAL DATABASE;
RMAN> BACKUP VALIDATE DATABASE;
RMAN> BACKUP VALIDATE DATABASE ARCHIVELOG ALL;

RMAN> BACKUP TABLESPACE test;
RMAN> BACKUP TABLESPACE system, users, tools;
RMAN> BACKUP TABLESPACE 4;
RMAN> BACKUP TABLESPACE gld PLUS ARCHIVELOG;
RMAN> BACKUP TABLESPACE invd INCLUDE CURRENT CONTROLFILE;
RMAN> BACKUP TABLESPACE appsd INCLUDE CURRENT CONTROLFILE PLUS ARCHIVELOG;
RMAN> BACKUP TABLESPACE dwh SECTION SIZE 100M;
RMAN> BACKUP SECTION SIZE 250M TABLESPACE datamart;
```

vahidusefzadeh@gmail.com

```
RMAN> BACKUP DATAFILE 1;
RMAN> BACKUP DATAFILE 3, 2, 14;
RMAN> BACKUP DATAFILE ‘/u01/data/...’;
RMAN> BACKUP DATAFILE 1 PLUS ARCHIVELOG;
RMAN> BACKUP KEEP FOREVER FORMAT '?/dbs/%U_longterm.cpy' TAG longterm_bck DATAFILE 1
DATAFILE 2;
RMAN> BACKUP SECTION SIZE 500M DATAFILE 6;

RMAN> BACKUP ARCHIVELOG ALL;
RMAN> BACKUP ARCHIVELOG ALL DELETE INPUT;
RMAN> BACKUP ARCHIVELOG LIKE '/arch%' DELETE ALL INPUT;
RMAN> BACKUP ARCHIVELOG FROM TIME ‘SYSDATE–3’;
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 100;
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 999 DELETE INPUT;
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 123 DELETE ALL INPUT;
RMAN> BACKUP ARCHIVELOG FROM SEQUENCE 21531 UNTIL SEQUENCE 21590 FORMAT
'/tmp/archive_backup.bkp';
RMAN> BACKUP ARCHIVELOG ALL FROM SEQUENCE 1200 DELETE ALL INPUT;
RMAN> BACKUP ARCHIVELOG NOT BACKED UP 2 TIMES;
RMAN> BACKUP ARCHIVELOG COMPLETION TIME BETWEEN 'SYSDATE-28' AND 'SYSDATE-7';
RMAN> BACKUP FORMAT='AL_%d/%t/%s/%p' ARCHIVELOG LIKE '%arc_dest%';

RMAN> BACKUP CURRENT CONTROLFILE;
OR
RMAN> SQL “ALTER DATABASE BACKUP CONTROLFILE TO ’’/u01/ .../bkctl.ctl’’ “;

RMAN> BACKUP CURRENT CONTROLFILE TO '/backup/cntrlfile.copy';
RMAN> BACKUP CONTROLFILE COPY ‘/u10/backup/control.bkp’;
RMAN> BACKUP SPFILE;
RMAN> BACKUP DEVICE TYPE sbt SPFILE ARCHIVELOG ALL;
RMAN> BACKUP DEVICE TYPE sbt DATAFILECOPY ALL NODUPLICATES;
```

RMAN> BACKUP RECOVERY FILES;

BACKUP set

RMAN> BACKUP BACKUPSET ALL;

RMAN> BACKUP BACKUPSET ALL FORMAT = ‘/u01/.../backup_%u.bak’;

RMAN> BACKUP BACKUPSET COMPLETED BEFORE ‘SYSDATE-3’ DELETE INPUT;

RMAN> BACKUP DEVICE TYPE sbt BACKUPSET COMPLETED BEFORE 'SYSDATE-14' DELETE INPUT;

RMAN> BACKUP COPIES 2 DEVICE TYPE sbt BACKUPSET ALL;

RMAN> BACKUP AS COMPRESSED BACKUPSET;

RMAN> BACKUP AS COMPRESSED BACKUPSET DEVICE TYPE DISK COPIES 2 DATABASE FORMAT '/disk1/db_%U', '/disk2/db_%U';

RMAN> BACKUP AS COMPRESSED BACKUPSET INCREMENTAL FROM SCN 4111140000000 DATABASE TAG 'RMAN_RECOVERY';

RMAN> BACKUP AS BACKUPSET DATAFILE '$ORACLE_HOME/oradata/users01.dbf','$ORACLE_HOME/oradata/tools01.dbf';

RMAN> BACKUP AS BACKUPSET DATAFILECOPY ALL;

RMAN> BACKUP AS BACKUPSET DATAFILECOPY ALL NODUPLICATES;

IMAGE copy

RMAN> BACKUP AS COPY DATABASE;

RMAN> BACKUP AS COPY COPY OF DATABASE FROM TAG 'test' CHECK LOGICAL TAG 'duptest';

RMAN> BACKUP AS COPY TABLESPACE 8;

RMAN> BACKUP AS COPY TABLESPACE test;

RMAN> BACKUP AS COPY TABLESPACE system, tools, users, undotbs;

RMAN> BACKUP AS COPY DATAFILE 1;

RMAN> BACKUP AS COPY DATAFILE 2 FORMAT '/disk2/df2.cpy' TAG my_tag;

RMAN> BACKUP AS COPY CURRENT CONTROLFILE;

RMAN> BACKUP AS COPY CURRENT CONTROLFILE FORMAT ‘/....’;

RMAN> BACKUP AS COPY ARCHIVELOG ALL;

RMAN> BACKUP AS COPY KEEP FOREVER NOLOGS CURRENT CONTROLFILE FORMAT '?/oradata/cf_longterm.cpy', DATAFILE 1 FORMAT '?/oradata/df1_longterm.cpy', DATAFILE 2 FORMAT

'?/oradata/df2_longterm.cpy';

RMAN> BACKUP AS COPY DATAFILECOPY 'bar' FORMAT 'foobar';

RMAN> BACKUP AS COPY DATAFILECOPY '/disk2/df2.cpy' FORMAT '/disk1/df2.cpy';

RMAN> BACKUP AS COPY REUSE TARGETFILE '/u01/oracle/11.2.0.2/dbs/orapwcrd' AUXILIARY FORMAT '/u01/oracle/11.2.0.2/dbs/orapwcrd';

RMAN> BACKUP AS COPY CURRENT CONTROLFILE FOR STANDBY AUXILIARY format '+DATA/crd/data1/control01.ctl';

Incremental backups

RMAN> BACKUP INCREMENTAL LEVEL=0 DATABASE;

RMAN> BACKUP INCREMENTAL LEVEL=1 DATABASE;

RMAN> BACKUP INCREMENTAL LEVEL=0 DATABASE PLUS ARCHIVELOG;

RMAN> BACKUP INCREMENTAL LEVEL 1 CUMULATIVE SKIP INACCESSIBLE DATABASE;

RMAN> BACKUP INCREMENTAL LEVEL 1 FOR RECOVER OF COPY WITH TAG 'incr' DATABASE;

RMAN> BACKUP DEVICE TYPE DISK INCREMENTAL LEVEL 1 FOR RECOVER OF COPY WITH TAG 'oltp' DATABASE;

RMAN> BACKUP DEVICE TYPE DISK INCREMENTAL FROM SCN 351986 DATABASE FORMAT '/tmp/incr_standby_%U';

RMAN> BACKUP INCREMENTAL FROM SCN 629184 DATAFILE 5 FORMAT '/tmp/ForStandby_%U' TAG 'FORSTANDBY';

RMAN> BACKUP INCREMENTAL LEVEL = --- tablespace/datafile

RMAN> BACKUP BLOCKS ALL CHECK LOGICAL VALIDATE DATAFILE 1398;

LIST command

Produce a detailed listing of backup sets or copies.

LIST

{ INCARNATION [OF DATABASE [[']database_name[']]]

| [EXPIRED] {listObjectSpec

vahidusefzadeh@gmail.com

[maintQualifier | RECOVERABLE [untilClause]]... | recordSpec}

};

listObjectSpec::=

{BACKUP [OF listObjectList] [listBackupOption] | COPY [OF listObjectList] | archivelogRecordSpecifier}

listObjectList::=

[DATAFILE datafileSpec [, datafileSpec]...

| TABLESPACE ['']tablespace_name['] [, ['']tablespace_name['']]...

| archivelogRecordSpecifier

| DATABASE [SKIP TABLESPACE ['']tablespace_name['] [, ['']tablespace_name['']] ...]

| CONTROLFILE

| SPFILE

]...

listBackupOption::=

[[BY BACKUP] [VERBOSE] | SUMMARY | BY {BACKUP SUMMARY|FILE}]

RMAN> LIST INCARNATION;

RMAN> LIST INCARNATION OF DATABASE;

RMAN> LIST INCARNATION OF DATABASE vis;

RMAN> LIST DB_UNIQUE_NAME ALL;

RMAN> LIST DB_UNIQUE_NAME OF DATABASE;

RMAN> LIST BACKUP;

RMAN> LIST BACKUP SUMMARY;

RMAN> LIST BACKUP BY FILE;

RMAN> LIST BACKUP OF DATABASE;

RMAN> LIST BACKUP OF DATABASE BY BACKUP;

RMAN> LIST BACKUP OF TABLESPACE test SUMMARY;

RMAN> LIST BACKUP OF DATAFILE 65;

RMAN> LIST BACKUP OF DATAFILE 11 SUMMARY;

```
RMAN> LIST BACKUP OF CONTROLFILE;
RMAN> LIST BACKUP OF ARCHIVELOG FROM SEQUENCE 2222;
RMAN> LIST BACKUP OF ARCHIVELOG FROM TIME 'sysdate-1';
RMAN> LIST BACKUP OF ARCHIVELOG ALL COMPLETED BEFORE 'sysdate-2';
RMAN> LIST BACKUP RECOVERABLE;
RMAN> LIST EXPIRED BACKUP;
RMAN> LIST EXPIRED BACKUP OF ARCHIVELOG ALL SUMMARY;

RMAN> LIST COPY;
RMAN> LIST COPY OF DATABASE ARCHIVELOG ALL;
RMAN> LIST COPY OF TABLESPACE appl_idx;
RMAN> LIST COPY OF DATAFILE 11, 60, 98;
RMAN> LIST COPY OF CONTROLFILE;
RMAN> LIST EXPIRED COPY;

RMAN> LIST BACKUPSET SUMMARY;
RMAN> LIST BACKUPSET 109;
RMAN> LIST BACKUPSET OF DATAFILE 1;
RMAN> LIST ARCHIVELOG;
RMAN> LIST ARCHIVELOG ALL LIKE '%5515%';
RMAN> LIST CONTROLFILECOPY "/tmp/cntrlfile.copy";

RMAN> LIST SCRIPT NAMES;
RMAN> LIST ALL SCRIPT NAMES;
RMAN> LIST GLOBAL SCRIPT NAMES;

RMAN> LIST FAILURE;
RMAN> LIST FAILURE 420 DETAIL;
RMAN> LIST FAILURE ALL;


RMAN> LIST RESTORE POINT ALL;
```

vahidusefzadeh@gmail.com

REPORT command

Report backup status: database, files, and backups. Perform detailed analyses of the content of the recovery catalog.

```
REPORT
{{NEED BACKUP [{INCREMENTAL | DAYS} [=] integer| REDUNDANCY [=] integer | RECOVERY WINDOW
OF integer DAYS)]
| UNRECOVERABLE
}
reportObject
| SCHEMA [atClause]
| OBSOLETE [obsOperandList]
}
[DEVICE TYPE deviceSpecifier [,deviceSpecifier]... ]

reportObject::=
[ DATAFILE datafileSpec [, datafileSpec]...
| TABLESPACE ']tablespace_name['] [, ']tablespace_name[']] ...
| DATABASE [SKIP TABLESPACE ']tablespace_name['] [, ']tablespace_name[']] ...]
]

atClause::=
{AT TIME [=] 'date_string' | AT SCN [=] integer|AT SEQUENCE [=] integer THREAD [=] integer
}

obsOperandList::=
[REDUNDANCY [=] integer | RECOVERY WINDOW OF integer DAYS | ORPHAN]...

RMAN> REPORT OBSOLETE;
```

RMAN> REPORT NEED BACKUP;

RMAN> REPORT NEED BACKUP DAYS=5;

RMAN> REPORT NEED BACKUP REDUNDANCY=3;

RMAN> REPORT NEED BACKUP RECOVERY WINDOW OF 7 DAYS;

RMAN> REPORT NEED BACKUP DATABASE;

RMAN> REPORT NEED BACKUP INCREMENTAL 1;

RMAN> REPORT UNRECOVERABLE;

RMAN> REPORT SCHEMA;

RMAN> REPORT SCHEMA AT TIME 'sysdate-20/1440';

CHANGE command

Update the status of a backup in the RMAN repository. Mark a backup piece, image copy, or archived redo log as having the status UNAVAILABLE or AVAILABLE; remove the repository record for a backup or copy; override the retention policy for a backup or copy; update the recovery catalog with the DB_UNIQUE_NAME for the target database.

CHANGE {BACKUP | COPY} [OF listObjList] [maintQualifier [maintQualifier]...]
{AVAILABLE | UNAVAILABLE | UNCATALOG | keepOption}
[DEVICE TYPE deviceSpecifier [, deviceSpecifier]...];

CHANGE archivelogRecordSpecifier {AVAILABLE | UNAVAILABLE | UNCATALOG | keepOption}
[DEVICE TYPE deviceSpecifier [, deviceSpecifier]...];

CHANGE recordSpec [DEVICE TYPE deviceSpecifier [, deviceSpecifier]...
{AVAILABLE | UNAVAILABLE | UNCATALOG | keepOption}
[DEVICE TYPE deviceSpecifier [, deviceSpecifier]...];

listObjList::=
[DATAFILE datafileSpec [, datafileSpec]...

| TABLESPACE ['] tablespace_name ['] [, ['] tablespace_name ']]...
| archivelogRecordSpecifier
| DATABASE [SKIP TABLESPACE ']tablespace_name['] [, ']tablespace_name['']] ...]
| CONTROLFILE
| SPFILE
]...

recordSpec::=
{{BACKUPPIECE | PROXY}
{'media_handle' [, 'media_handle']... | primary_key [, primary_key]... | TAG [=] ['] tag_name ']
}
| BACKUPSET primary_key [, primary_key]...
| {CONTROLFILECOPY | DATAFILECOPY}
{{primary_key [, primary_key]... | 'filename' [, 'filename']...}
| TAG [=] ['] tag_name ['] [, ['] tag_name ']]...
}
| ARCHIVELOG {primary_key [, primary_key]... | 'filename' [, 'filename']...}
}

RMAN> CHANGE BACKUPSET 666 KEEP FOREVER;
RMAN> CHANGE BACKUPSET 431 KEEP FOREVER NOLOGS;
RMAN> CHANGE BACKUPSET 100 UNAVAILABLE;
RMAN> CHANGE BACKUPSET 123 NOKEEP;
RMAN> CHANGE BACKUPSET 121,122,127,203,300 UNCATALOG;
RMAN> CHANGE BACKUP OF DATABASE TAG=’abc’ UNAVAILABLE;
RMAN> CHANGE BACKUP OF DATABASE DEVICE TYPE DISK UNAVAILABLE;
RMAN> CHANGE COPY OF DATABASE CONTROLFILE NOKEEP;
RMAN> CHANGE BACKUP OF SPFILE COMPLETED BEFORE 'SYSDATE-3' UNAVAILABLE;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' KEEP FOREVER;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' DATABASE KEEP FOREVER;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' KEEP FOREVER NOLOGS;
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup' NOKEEP;

RMAN> CHANGE ARCHIVELOG ALL UNCATALOG;

RMAN> CHANGE CONTROLFILECOPY '/tmp/cf.cpy' UNCATALOG;

RMAN> CHANGE FAILURE 5 PRIORITY LOW;

RMAN> CHANGE BACKUP FOR DB_UNIQUE_NAME standby1 RESET DB_UNIQUE_NAME;

RMAN> CHANGE BACKUP FOR DB_UNIQUE_NAME standby3 RESET DB_UNIQUE_NAME TO standby2;

RMAN> CHANGE DB_UNIQUE_NAME FROM rdbms4 TO rdbms_dev;

CROSSCHECK command

Check whether files managed by RMAN, such as archived logs, datafile copies, and backup pieces, still exist on disk or tape.

CROSSCHECK

{{BACKUP [OF listObjList] | COPY [OF listObjList] | archivelogRecordSpecifier} [maintQualifier

[maintQualifier]...]

| recordSpec [DEVICE TYPE deviceSpecifier [, deviceSpecifier]...]

};

listObjList::=

[DATAFILE datafileSpec [, datafileSpec]...

| TABLESPACE ['] tablespace_name ['] [, ['] tablespace_name ']]...

| archivelogRecordSpecifier

| DATABASE [SKIP TABLESPACE ']tablespace_name['] [, ']tablespace_name[']] ...]

| CONTROLFILE

| SPFILE

]...

recordSpec::=

{{ BACKUPPIECE | PROXY }

vahidusefzadeh@gmail.com

{ 'media_handle' [, 'media_handle']...| primary_key [, primary_key]... | TAG [=] ['] tag_name ['] }
| BACKUPSET primary_key [, primary_key]...
| { CONTROLFILECOPY | DATAFILECOPY }
{ {primary_key [, primary_key]... | 'filename' [, 'filename']...}
| TAG [=] ['] tag_name ['] [, ['] tag_name ']]...
}
| ARCHIVELOG { primary_key [, primary_key]... | 'filename' [, 'filename']... }
}

RMAN> CROSSCHECK BACKUP;
RMAN> CROSSCHECK BACKUP TAG='full db';
RMAN> CROSSCHECK BACKUP COMPLETED BETWEEN 'SYSDATE-7' AND 'SYSDATE–1';
RMAN> CROSSCHECK BACKUP COMPLETED BETWEEN '01-JAN-10' AND '14-FEB-10';
RMAN> CROSSCHECK BACKUP DEVICE TYPE sbt COMPLETED BETWEEN '01-AUG-09' AND '31-DEC-09';
RMAN> CROSSCHECK BACKUP DEVICE TYPE DISK COMPLETED BETWEEN '01-JAN-10' AND '23-MAR-10';

RMAN> CROSSCHECK BACKUP OF DATABASE;
RMAN> CROSSCHECK BACKUP OF TABLESPACE warehouse;
RMAN> CROSSCHECK BACKUP OF TABLESPACE userd COMPLETED BEFORE 'SYSDATE-14';
RMAN> CROSSCHECK BACKUP OF TABLESPACES gld, invd;
RMAN> CROSSCHECK BACKUP OF DATAFILE 9;
RMAN> CROSSCHECK BACKUP OF DATAFILE 4 COMPLETED AFTER 'SYSDATE-14';
RMAN> CROSSCHECK BACKUP OF DATAFILE "?/oradata/dwh/system01.dbf" COMPLETED AFTER
'SYSDATE-14';
RMAN> CROSSCHECK BACKUP OF CONTROLFILE;
RMAN> CROSSCHECK BACKUP OF SPFILE;
RMAN> CROSSCHECK BACKUP OF ARCHIVELOG ALL;
RMAN> CROSSCHECK BACKUP OF ARCHIVELOG ALL SPFILE;

RMAN> CROSSCHECK COPY;
RMAN> CROSSCHECK COPY OF DATABASE;
RMAN> CROSSCHECK DATAFILECOPY 113, 114, 115;

RMAN> CROSSCHECK CONTROLFILECOPY '/tmp/control01.ctl';

RMAN> CROSSCHECK ARCHIVELOG ALL;

RMAN> CROSSCHECK BACKUPSET;

RMAN> CROSSCHECK BACKUPSET 1338, 1339, 1340;

RMAN> CROSSCHECK BACKUPPIECE TAG = 'nightly_backup';

RMAN> CROSSCHECK PROXY 789;

SQL command

Execute a SQL statement from within Recovery Manager.

SQL [CHANNEL ‘channel_id’] ‘command’;

RMAN> SQL 'ALTER TABLESPACE users ONLINE';

RMAN> SQL 'ALTER DATABASE DATAFILE 64 OFFLINE';

RMAN> SQL "ALTER SYSTEM ARCHIVE LOG CURRENT";

RMAN> SQL "ALTER SYSTEM SWITCH LOGFILE";

RMAN> SQL "ALTER DATABASE BACKUP CONTROLFILE TO TRACE";

RMAN> SQL "ALTER TABLESPACE users ADD DATAFILE ''/disk1/ora/users02.dbf'' SIZE 100K AUTOEXTEND ON NEXT 10K MAXSIZE 100K";

RESTORE command

Restore files from backup sets or from disk copies to the default or a new location.

RESTORE

[(] restoreObject [(restoreSpecOperand [restoreSpecOperand]...] [)]...

[CHANNEL ['] channel_id ']

| PARMS [=] 'channel_parms'

| FROM { BACKUPSET | DATAFILECOPY }

```
| untilClause
| FROM TAG [=] ['] tag_name ']
| VALIDATE
| DEVICE TYPE deviceSpecifier [, deviceSpecifier]...
]...;

restoreObject::=
{ CONTROLFILE [TO 'filename']
| DATABASE [SKIP [FOREVER] TABLESPACE ']tablespace_name['] [, ']tablespace_name[']] ...]
| DATAFILE datafileSpec [, datafileSpec]...
| TABLESPACE ['] tablespace_name ['] [, '] tablespace_name ']]...
| archivelogRecordSpecifier
| SPFILE [TO [PFILE] 'filename']
}

restoreSpecOperand::=
{ CHANNEL ['] channel_id ['] | FROM TAG [=] ['] tag_name ['] | PARMS [=] 'channel_parms'
| FROM {AUTOBACKUP [{MAXSEQ | MAXDAYS} [=] integer)]... | 'media_handle'}
}

RMAN> RESTORE DATABASE;
RMAN> RESTORE DATABASE VALIDATE;
RMAN> RESTORE DATABASE PREVIEW;
RMAN> RESTORE DATABASE PREVIEW SUMMARY;
RMAN> RESTORE DATABASE SKIP TABLESPACE temp, history;
RMAN> RESTORE DATABASE UNTIL SCN 154876;

RMAN> RESTORE TABLESPACE users;
RMAN> RESTORE TABLESPACE dwh1, dwh2;
RMAN> RESTORE TABLESPACE tbs1 PREVIEW;
RMAN> RESTORE TABLESPACE users VALIDATE;
```

```
RMAN> RESTORE DATAFILE 45;
RMAN> RESTORE DATAFILE 23 PREVIEW;
RMAN> RESTORE DATAFILE 12 VALIDATE;

RMAN> RESTORE CONTROLFILE;
RMAN> RESTORE CONTROLFILE FROM AUTOBACKUP;
RMAN> RESTORE CONTROLFILE FROM TAG 'monday_cf_backup';
RMAN> RESTORE CONTROLFILE FROM '/u01/control01.ctl';
RMAN> RESTORE CONTROLFILE VALIDATE;
RMAN> RESTORE CONTROLFILE TO '/tmp/autobkp.dbf' FROM AUTOBACKUP MAXSEQ 20 MAXDAYS 150;

RMAN> RESTORE SPFILE;
RMAN> RESTORE SPFILE FROM AUTOBACKUP;
RMAN> RESTORE ARCHIVELOG ALL VALIDATE;
RMAN> RESTORE ARCHIVELOG ALL PREVIEW;
RMAN> RESTORE ARCHIVELOG ALL PREVIEW RECALL;
RMAN> RESTORE ARCHIVELOG ALL DEVICE TYPE sbt PREVIEW;
RMAN> RESTORE ARCHIVELOG LOW LOGSEQ 78311 HIGH LOGSEQ 78340 THREAD 1 ALL;
RMAN> RESTORE ARCHIVELOG FROM LOGSEQ=21531 UNTIL LOGSEQ=21590;
RMAN> RESTORE STANDBY CONTROLFILE FROM TAG 'forstandby';
RMAN> RESTORE CLONE CONTROLFILE TO '+DATA/pcrd/data2/control02.ctl' FROM
'+DATA/pcrd/data1/control01.ctl';
```

Restore the control file, (to all locations specified in the parameter file) then restore the database, using that control file:

```
STARTUP NOMOUNT;
RUN
{
ALLOCATE CHANNEL c1 DEVICE TYPE sbt;
RESTORE CONTROLFILE;
ALTER DATABASE MOUNT;
RESTORE DATABASE;
```

}

RECOVER command

Perform media recovery from RMAN backups and copies. Apply redo log files and incremental backups to datafiles or data blocks restored from backup or datafile copies, to update them to a specified time.

```
RECOVER [DEVICE TYPE deviceSpecifier [, deviceSpecifier]...]
recoverObject [recoverOptionList];

recoverObject::=
{ DATABASE
[ untilClause
| [untilClause] SKIP [FOREVER] TABLESPACE ']tablespace_name[' [, ']tablespace_name[']] ...]
| TABLESPACE ']tablespace_name[' [, ']tablespace_name[']]...
| DATAFILE datafileSpec [, datafileSpec]...
}

recoverOptionList::=
{ DELETE ARCHIVELOG [MAXSIZE {integer [K | M | G]}]
| CHECK READONLY
| NOREDO
| {FROM TAG | ARCHIVELOG TAG} [=] ['] tag_name ['
...
}
```

```
RMAN> RECOVER DATABASE;
RMAN> RECOVER DATABASE NOREDO;
RMAN> RECOVER DATABASE SKIP TABLESPACE temp;
RMAN> RECOVER DATABASE SKIP FOREVER TABLESPACE exam;
```

RMAN> RECOVER DATABASE UNTIL SCN 154876;

RMAN> RECOVER TABLESPACE users;

RMAN> RECOVER TABLESPACE dwh DELETE ARCHIVELOG MAXSIZE 2M;

RMAN> RECOVER DATAFILE 33;

RMAN> RECOVER DATAFILE 3 BLOCK 116 DATAFILE 4 BLOCK 10;

RMAN> RECOVER DATAFILE 2 BLOCK 204 DATAFILE 9 BLOCK 109 FROM TAG=sundaynight;

RMAN> RECOVER DATAFILECOPY '/disk1/img.df' UNTIL TIME 'SYSDATE-7';

RMAN> RECOVER COPY OF DATABASE WITH TAG 'incr';

RMAN> RECOVER COPY OF DATABASE WITH TAG 'upd' UNTIL TIME 'SYSDATE - 7';

RMAN> RECOVER CORRUPTION LIST;

Restore and recover the whole database

RMAN> STARTUP FORCE MOUNT;

RMAN> RESTORE DATABASE;

RMAN> RECOVER DATABASE;

RMAN> ALTER DATABASE OPEN;

Restore and recover a tablespace

RMAN> SQL 'ALTER TABLESPACE users OFFLINE';

RMAN> RESTORE TABLESPACE users;

RMAN> RECOVER TABLESPACE users;

RMAN> SQL 'ALTER TABLESPACE users ONLINE';

Restore and recover a datafile

RMAN> SQL 'ALTER DATABASE DATAFILE 64 OFFLINE';

RMAN> RESTORE DATAFILE 64;

RMAN> RECOVER DATAFILE 64;

RMAN> SQL 'ALTER DATABASE DATAFILE 64 ONLINE';

Steps for media recovery:

1. Mount or open the Oracle database. Mount the database when performing whole database recovery,

or open the database when performing online tablespace/datafile recovery.

2. To perform incomplete recovery, use the SET UNTIL command to specify the time, SCN, or log sequence number at which recovery terminates. Alternatively, specify the UNTIL clause on the RESTORE and RECOVER commands.

3. Restore the necessary files with the RESTORE command.

4. Recover the datafiles with the RECOVER command.

5. Place the database in its normal state. For example, open it or bring recovered tablespaces/datafiles online.

DELETE command

Delete backups and copies, remove references to them from the recovery catalog, and update their control file records to status DELETED.

```
DELETE [FORCE] [NOPROMPT]
{[EXPIRED]
{
{BACKUP [OF listObjectList] | COPY [OF listObectjList] | archivelogRecordSpecifier
} [maintQualifier [maintQualifier]...]
| recordSpec [DEVICE TYPE deviceSpecifier [, deviceSpecifier]...]
}
| OBSOLETE [REDUNDANCY [=] integer | RECOVERY WINDOW OF integer DAYS | ORPHAN] [DEVICE TYPE
(deviceSpecifier [, deviceSpecifier]...]
};

recordSpec::=
{ { BACKUPPIECE | PROXY }
{ 'media_handle' [, 'media_handle']...| primary_key [, primary_key]...| TAG [=] ['] tag_name ['] }
| BACKUPSET primary_key [, primary_key]...
| { CONTROLFILECOPY | DATAFILECOPY }
```

vahidusefzadeh@gmail.com

{ {primary_key [, primary_key]... | 'filename' [, 'filename']...}
| TAG [=] ['] tag_name ['] [, ['] tag_name ']]...
}
| ARCHIVELOG { primary_key [, primary_key]... | 'filename' [, 'filename']... }

listObjectList::=
[DATAFILE datafileSpec [, datafileSpec]...
| TABLESPACE ['] tablespace_name ['] [, ['] tablespace_name ']]...
| archivelogRecordSpecifier
| DATABASE [SKIP TABLESPACE ']tablespace_name['] [, ']tablespace_name['']] ...]
| CONTROLFILE
| SPFILE
]...

RMAN> DELETE OBSOLETE;
RMAN> DELETE NOPROMPT OBSOLETE;
RMAN> DELETE NOPROMPT OBSOLETE RECOVERY WINDOW OF 7 DAYS;
RMAN> DELETE EXPIRED BACKUP;
RMAN> DELETE EXPIRED BACKUP DEVICE TYPE sbt;
RMAN> DELETE BACKUP OF DATABASE LIKE '/tmp%';
RMAN> DELETE NOPROMPT EXPIRED BACKUP OF TABLESPACE userd COMPLETED BEFORE 'SYSDATE-14';
RMAN> DELETE BACKUP OF SPFILE TABLESPACE users DEVICE TYPE SBT;

RMAN> DELETE ARCHIVELOG ALL;
RMAN> DELETE ARCHIVELOG ALL COMPLETED BEFORE 'sysdate-2';
RMAN> DELETE ARCHIVELOG ALL BACKED UP 2 TIMES TO DEVICE TYPE SBT;
RMAN> DELETE ARCHIVELOG ALL LIKE '%755153075%';
RMAN> DELETE ARCHIVELOG UNTIL SEQUENCE=79228;
RMAN> DELETE FORCE ARCHIVELOG ALL COMPLETED BEFORE 'sysdate-1.5';
RMAN> DELETE FORCE ARCHIVELOG UNTIL SEQUENCE=16364;
RMAN> DELETE NOPROMPT ARCHIVELOG UNTIL SEQUENCE = 7300;
RMAN> DELETE EXPIRED ARCHIVELOG ALL;

www.usefzadeh.com

RMAN> DELETE NOPROMPT EXPIRED ARCHIVELOG ALL;

RMAN> DELETE BACKUPSET 101, 102, 103;

RMAN> DELETE NOPROMPT BACKUPSET TAG weekly_bkup;

RMAN> DELETE FORCE NOPROMPT BACKUPSET TAG weekly_bkup;

RMAN> DELETE DATAFILECOPY "+DG_DATA/db/datafile/system.259.699468079";

RMAN> DELETE CONTROLFILECOPY '/tmp/cntrlfile.copy';

RMAN> DELETE BACKUP DEVICE TYPE SBT;

RMAN> DELETE BACKUP DEVICE TYPE DISK;

RMAN> DELETE COPY;

RMAN> DELETE EXPIRED COPY;

RMAN> DELETE COPY TAG 'lastest';

DROP DATABASE command

Delete the target database from disk and unregisters it.

RMAN> DROP DATABASE;

RMAN> DROP DATABASE NOPROMPT;

RMAN> DROP DATABASE INCLUDING BACKUPS;

RMAN> DROP DATABASE INCLUDING BACKUPS NOPROMPT;

DUPLICATE command

Use backups of the target database to create a duplicate database that we can use for testing purposes or to create a standby database.

RMAN> DUPLICATE TARGET DATABASE;

RMAN> DUPLICATE TARGET DATABASE TO dwhdb;

RMAN> DUPLICATE TARGET DATABASE TO test PFILE=/u01/apps/db/inittest.ora;

```
RMAN> DUPLICATE TARGET DATABASE TO devdb NOFILENAMECHECK;
RMAN> DUPLICATE DATABASE 'prod' DBID 139525561 TO 'dupdb' NOFILENAMECHECK;
RMAN> DUPLICATE DATABASE TO "cscp" NOFILENAMECHECK BACKUP LOCATION '/apps/oracle/backup';
RMAN> DUPLICATE TARGET DATABASE TO dup FROM ACTIVE DATABASE NOFILENAMECHECK
PASSWORD FILE SPFILE;

RMAN> DUPLICATE TARGET DATABASE TO dupdb
  LOGFILE GROUP 1 ('?/dbs/dupdb_log_1_1.f','?/dbs/dupdb_log_1_2.f') SIZE 200K, GROUP 2
('?/dbs/dupdb_log_2_1.f','?/dbs/dupdb_log_2_2.f') SIZE 200K REUSE;
RMAN> DUPLICATE TARGET DATABASE TO dup FOR STANDBY FROM ACTIVE DATABASE PASSWORD FILE
SPFILE PARAMETER_VALUE_CONVERT '/disk1', '/disk2'
  SET DB_FILE_NAME_CONVERT '/disk1','/disk2'
  SET LOG_FILE_NAME_CONVERT '/disk1','/disk2'
  SET SGA_MAX_SIZE 200M SET SGA_TARGET 125M;


RMAN> DUPLICATE TARGET DATABASE FOR STANDBY NOFILENAMECHECK;
RMAN> DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE;
RMAN> DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE NOFILENAMECHECK;
RMAN> DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE
  SPFILE PARAMETER_VALUE_CONVERT '/stg/','/muc/'
  SET "DB_UNIQUE_NAME"="muc"
  SET LOG_FILE_NAME_CONVERT '/stg/','/muc/'
  SET DB_FILE_NAME_CONVERT '/stg/','/muc/'
  DORECOVER;
RMAN> DUPLICATE DATABASE TO newdb
  UNTIL RESTORE POINT firstquart12
  DB_FILE_NAME_CONVERT='/u01/prod1/dbfiles/','/u01/newdb/dbfiles'
  PFILE = '/u01/newdb/admin/init.ora';
```

SWITCH command

Specify that a datafile copy is now the current datafile, i.e. the datafile pointed to by the control file. This command is equivalent to the SQL statement ALTER DATABASE RENAME FILE as it applies to datafiles.

RMAN> SWITCH DATABASE TO COPY;

RMAN> SWITCH TABLESPACE users TO COPY;

RMAN> SWITCH DATAFILE ALL;

RMAN> SWITCH DATAFILE '/disk1/tols.dbf' TO DATAFILECOPY '/disk2/tols.copy';

RMAN> SWITCH DATAFILE "+DG_OLD/db/datafile/sysaux.260.699468081" TO COPY;

RMAN> SWITCH TEMPFILE 1;

RMAN> SWITCH TEMPFILE 1 TO '/newdisk/dbs/temp1.f';

RMAN> SWITCH TEMPFILE ALL;

RMAN> SWITCH CLONE DATAFILE ALL;

CATALOG command

Add information about file copies and user-managed backups to the catalog repository.

RMAN> CATALOG DATAFILECOPY '/disk1/old_datafiles/01_10_2009/users01.dbf';

RMAN> CATALOG DATAFILECOPY '/disk2/backup/users01.bkp' LEVEL 0;

RMAN> CATALOG CONTROLFILECOPY '/disk3/backup/cf_copy.bkp';

RMAN> CATALOG ARCHIVELOG '/disk1/arch1_731.dbf', '/disk1/arch1_732.dbf';

RMAN> CATALOG BACKUPPIECE '/disk1/c-874220581-20090428-01';

RMAN> CATALOG LIKE '/backup';

RMAN> CATALOG START WITH '/fs2/arch';

RMAN> CATALOG START WITH '/disk2/archlog' NOPROMPT;

RMAN> CATALOG START WITH '+dg2';

RMAN> CATALOG RECOVERY AREA;

ALLOCATE command

Establish a channel, which is a connection between RMAN and a database instance.

RMAN> ALLOCATE CHANNEL c1 DEVICE TYPE sbt;

RMAN> ALLOCATE CHANNEL ch DEVICE TYPE DISK FORMAT ‘C:\ORACLEBKP\DB_U%’;

RMAN> ALLOCATE CHANNEL t1 DEVICE TYPE DISK CONNECT 'sys/pwd@bkp1’;

RMAN> ALLOCATE CHANNEL c1 DEVICE TYPE sbt PARMS 'ENV=(OB_MEDIA_FAMILY=wholedb_mf)';

RMAN> ALLOCATE CHANNEL t1 DEVICE TYPE sbt PARMS 'ENV=(OB_DEVICE_1=tape1,
OB_DEVICE_2=tape3)';

RMAN> ALLOCATE CHANNEL t1 TYPE 'sbt_tape'
PARMS='SBT_LIBRARY=/usr/openv/netbackup/bin/libobk.so.1';

RMAN> ALLOCATE CHANNEL t1 TYPE 'sbt_tape' SEND "NB_ORA_CLIENT=CLIENT_MACHINE_NAME";

RMAN> ALLOCATE CHANNEL 'dev1' TYPE 'sbt_tape' PARMS 'ENV=OB2BARTYPE=ORACLE8,
OB2APPNAME=USEFDB, OB2BARLIST=MACHINENAME_USEFDB_ARCHLOGS)';

RMAN> ALLOCATE CHANNEL y1 TYPE DISK RATE 70M;

RMAN> ALLOCATE AUXILIARY CHANNEL ac1 TYPE DISK;

RMAN> ALLOCATE AUXILIARY CHANNEL ac2 DEVICE TYPE sbt;

ALLOCATE CHANNEL FOR MAINTENANCE - allocate a channel in preparation for issuing maintenance commands such as DELETE.

RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK;

RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK FORMAT "/disk2/%U";

RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE DISK CONNECT '@test1';

RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE sbt;

RMAN> ALLOCATE CHANNEL FOR MAINTENANCE DEVICE TYPE sbt PARMS
'SBT_LIBRARY=/usr/local/oracle/backup/lib/libobk.so, ENV=(OB_DEVICE_1=tape2)';

RELEASE CHANNEL command

Release a channel that was allocated with an ALLOCATE CHANNEL or ALLOCATE CHANNEL FOR MAINTENANCE command.

RMAN> RELEASE CHANNEL;

RMAN> RELEASE CHANNEL c1;

BLOCKRECOVER command

Will recover the corrupted blocks.

RMAN> BLOCKRECOVER CORRUPTION LIST;

RMAN> BLOCKRECOVER DATAFILE 8 BLOCK 22;

RMAN> BLOCKRECOVER DATAFILE 7 BLOCK 233,235 DATAFILE 4 BLOCK 101;

RMAN> BLOCKRECOVER DATAFILE 2 BLOCK 12,13 DATAFILE 3 BLOCK 5,98,99 DATAFILE 4 BLOCK 19;

RMAN> BLOCKRECOVER DATAFILE 3 BLOCK 2,4,5 TABLESPACE sales DBA 4194405,4194412 FROM DATAFILECOPY;

RMAN> BLOCKRECOVER TABLESPACE dwh DBA 4194404,4194405 FROM TAG "weekly";

RMAN> BLOCKRECOVER TABLESPACE dwh DBA 94404 RESTORE UNTIL TIME 'SYSDATE-2';

ADVISE FAILURE command (From Oracle 11g R1)

Display repair options.

RMAN> ADVISE FAILURE;

RMAN> ADVISE FAILURE 555, 242;

RMAN> ADVISE FAILURE ALL;

RMAN> ADVISE FAILURE CRITICAL;

RMAN> ADVISE FAILURE HIGH;

RMAN> ADVISE FAILURE LOW;

RMAN> ADVISE FAILURE HIGH EXCLUDE FAILURE 625;

REPAIR FAILURE command (From Oracle 11g R1)

Repair one or more failures recorded in the automated diagnostic repository.

RMAN> REPAIR FAILURE;

RMAN> REPAIR FAILURE PREVIEW;

RMAN> REPAIR FAILURE NOPROMPT;

RMAN> REPAIR FAILURE USING ADVISE OPTION integer;

VALIDATE command

Examine a backup set and report whether its data is intact. RMAN scans all of the backup pieces in the specified backup sets and looks at the checksums to verify that the contents can be successfully restored.

RMAN> VALIDATE BACKUPSET 218;

RMAN> VALIDATE BACKUPSET 3871, 3890;

RMAN> VALIDATE DATABASE;

RMAN> VALIDATE CHECK LOGICAL DATABASE;

RMAN> VALIDATE SKIP INACCESSIBLE DATABASE;

RMAN> VALIDATE COPY OF DATABASE;

RMAN> VALIDATE TABLESPACE dwh;

RMAN> VALIDATE COPY OF TABLESPACE dwh;

RMAN> VALIDATE DATAFILE 2;

RMAN> VALIDATE DATAFILE 4,8;

RMAN> VALIDATE DATAFILE 4 BLOCK 56;

RMAN> VALIDATE DATAFILE 8 SECTION SIZE = 200M;

RMAN> VALIDATE CURRENT CONTROLFILE;

RMAN> VALIDATE SPFILE;

RMAN> VALIDATE RECOVERY FILES;

RMAN> VALIDATE RECOVERY AREA;

```
RMAN> VALIDATE DB_RECOVERY_FILE_DEST;
```

SPOOL command

Write RMAN output to a log file.

```
RMAN> SPOOL LOG TO '/tmp/spool.log';
RMAN> SPOOL LOG TO '/tmp/backup.log' APPEND;
RMAN> SPOOL LOG OFF;
```

---

run command

Execute a sequence of one or more RMAN commands, which are one or more statements executed within the braces of RUN.

```
RMAN> run {
ALLOCATE CHANNEL c1 TYPE DISK FORMAT '/orabak/%U';
BACKUP TABLESPACE users;
}
RMAN> run {
  ALLOCATE CHANNEL c1 TYPE DISK FORMAT '&1/%U';
  BACKUP TABLESPACE &2;
}
RMAN> run
{
  ALLOCATE CHANNEL dev1 DEVICE TYPE DISK FORMAT '/fs1/%U';
  ALLOCATE CHANNEL dev2 DEVICE TYPE DISK FORMAT '/fs2/%U';
  BACKUP(TABLESPACE system,fin,mark FILESPERSET 20) (DATAFILE 2,3,6);
}
```

CREATE SCRIPT command

Create a stored script and store it in the recovery catalog.

```
RMAN> CREATE SCRIPT backup_whole
COMMENT "backup whole database and archived redo log files"
{BACKUP INCREMENTAL LEVEL 0 TAG backup_whole FORMAT "/disk2/backup/%U" DATABASE PLUS
ARCHIVELOG;}

RMAN> CREATE SCRIPT backup_ts_users
COMMENT 'tablespace users backup'
{ALLOCATE CHANNEL c1 TYPE DISK FORMAT 'c:\temp\%U';
BACKUP TABLESPACE users;}

RMAN> CREATE SCRIPT df {BACKUP DATAFILE &1 TAG &2.1 FORMAT '/disk1/&3_%U';}
RMAN> CREATE SCRIPT backup_ts_users FROM FILE 'backup_ts_users.rman';
RMAN> CREATE GLOBAL SCRIPT gl_backup_db {BACKUP DATABASE PLUS ARCHIVELOG;}
RMAN> CREATE GLOBAL SCRIPT backup_db
COMMENT "back up any database from the recovery catalog, with logs"
{BACKUP DATABASE;}
```

PRINT SCRIPT command

Display a stored script.

```
RMAN> PRINT SCRIPT backup_db;
RMAN> PRINT GLOBAL SCRIPT backup_db;
RMAN> PRINT GLOBAL SCRIPT gl_backup_db TO FILE "/tmp/gl_backupdb.rman";
```

---

vahidusefzadeh@gmail.com

REPLACE SCRIPT command

Replace an existing script stored in the recovery catalog. If the script does not exist, then REPLACE SCRIPT creates it.

RMAN> REPLACE SCRIPT backup_db {BACKUP DATABASE PLUS ARCHIVELOG;}

RMAN> REPLACE SCRIPT df {BACKUP DATAFILE &1 TAG &2.1 FORMAT '&3_%U';}

RMAN> REPLACE GLOBAL SCRIPT backup_db {BACKUP DATABASE PLUS ARCHIVELOG;}

RMAN> REPLACE GLOBAL SCRIPT gl_full_bkp FROM FILE '/tmp/script_file.txt';

EXECUTE SCRIPT command

Run an RMAN stored script.

RMAN> RUN {EXECUTE SCRIPT backup_whole;}

RMAN> RUN {EXECUTE SCRIPT backup_ts_any USING 'example';}

RMAN> RUN {EXECUTE SCRIPT backup_df USING 3 test_backup df3;}

RMAN> RUN {EXECUTE GLOBAL SCRIPT global_backup_db;}

DELETE SCRIPT command

Delete a stored script from the recovery catalog.

RMAN> DELETE SCRIPT backup_db;

RMAN> DELETE GLOBAL SCRIPT global_backup_db;

FLASHBACK DATABASE command

Return the database to its state at a previous time or SCN.

RMAN> FLASHBACK DATABASE TO SCN 411010;

RMAN> FLASHBACK DATABASE TO RESTORE POINT 'before_update';

TRANSPORT TABLESPACE command

Create transportable tablespace sets from backup for one or more tablespaces.

RMAN> TRANSPORT TABLESPACE example, tools

TABLESPACE DESTINATION '/disk1/trans' AUXILIARY DESTINATION '/disk1/aux' UNTIL TIME 'SYSDATE-15/1440';

RMAN> TRANSPORT TABLESPACE exam

TABLESPACE DESTINATION '/disk1/trans' AUXILIARY DESTINATION '/disk1/aux' DATAPUMP DIRECTORY dpdir DUMP FILE 'dmpfile.dmp' IMPORT SCRIPT 'impscript.sql' EXPORT LOG 'explog.log';

CONVERT command

Convert datafile formats for transporting tablespaces and databases across platforms.

RMAN> CONVERT DATABASE NEW DATABASE 'prodwin' TRANSPORT SCRIPT '/tmp/convertdb/transportscript' TO PLATFORM 'Microsoft Windows IA (32-bit)' DB_FILE_NAME_CONVERT '/disk1/oracle/dbs','/tmp/convertdb';

RMAN> CONVERT DATABASE ON DESTINATION PLATFORM CONVERT SCRIPT '/tmp/convertdb/convertscript.rman' TRANSPORT SCRIPT '/tmp/convertdb/transportscript.sql' NEW DATABASE 'prodwin' FORMAT '/tmp/convertdb/%U';

RMAN> CONVERT DATABASE ON DESTINATION PLATFORM CONVERT SCRIPT '/tmp/convert_newdb.rman' TRANSPORT SCRIPT '/tmp/transport_newdb.sql' NEW DATABASE 'prodaix' DB_FILE_NAME_CONVERT '/u01/oradata/datafile','+DATA';

RMAN> CONVERT TABLESPACE tbs_2 FORMAT '/tmp/tbs_2_%U.df';

RMAN> CONVERT TABLESPACE fin, hr TO PLATFORM 'Solaris[tm] OE (32-bit)';

RMAN> CONVERT TABLESPACE fin, hr TO PLATFORM 'Solaris[tm] OE (32-bit)' FORMAT '/tmp/transport_to_solaris/%U';

```
RMAN> CONVERT DATAFILE '/disk1/oracle/dbs/tbs_f1.df', '/disk1/oracle/dbs/ax1.f' FORMAT
'+DATAFILE';
RMAN> CONVERT DATAFILE '/u01/oradata/datafile/system.dbf' FROM PLATFORM 'Linux x86 64-bit'
FORMAT '+DATA/system.dbf';
RMAN> CONVERT DATAFILE
'/tmp/from_solaris/fin/fin01.dbf', '/tmp/from_solaris/fin/fin02.dbf',
'/tmp/from_solaris/hr/hr01.dbf', '/tmp/from_solaris/hr/hr02.dbf'
DB_FILE_NAME_CONVERT '/tmp/from_solaris/fin','/disk2/orahome/dbs/fin',
'/tmp/from_solaris/hr','/disk2/orahome/dbs/hr'
FROM PLATFORM 'Solaris[tm] OE (64-bit)';


RMAN> CONVERT DATAFILE '/tmp/PSMN.dbf' TO PLATFORM='Solaris Operating System (x86-64)' FROM
PLATFORM='Solaris[tm] OE (64-bit)'
FORMAT '/tmp/test/%N.dbf' DB_FILE_NAME_CONVERT='/ui/prod/oracle/oradata/SEARCHP/data/',
'/tmp/test';
```

EXIT or QUIT Command

Exit the RMAN console.

```
RMAN> exit;
RMAN> quit;
```

SEND command

Send a vendor-specific quoted string to one or more specific channels.

```
RMAN> SEND 'OB_DEVICE tape1';
```

HOST command

Invoke an operating system command-line subshell from within RMAN or run a specific operating system command.

```
RMAN> HOST;
RMAN> HOST 'ls -lt /disk2/*';
RMAN> HOST '/bin/mv $ORACLE_HOME/dbs/*.arc /disk2/archlog/';
```

vahidusefzadeh@gmail.com

chapter 6

data guard

--Enable Forced Logging(primary)

SQL> ALTER DATABASE FORCE LOGGING;

--create controlfile for standby

alter database create standby controlfile as '/u01/usefdb/ control01.ctl';

ALTER DATABASE CREATE FAR SYNC INSTANCE CONTROLFILE AS '/u01/usefdb/control01.ctl';

--create a standby redo log

ALTER DATABASE ADD STANDBY LOGFILE '<path_and_file_name>' SIZE <integer> <M | G>;

ALTER DATABASE ADD STANDBY LOGFILE ('/oracle/dbs/slog1.rdo') SIZE 500M;

ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 SIZE 500M;

select group#, bytes, 'online' as type from v$log union select group#, bytes,'standby' as type from

v$standby_log order by 1;

SELECT GROUP#,THREAD#,SEQUENCE#,ARCHIVED,STATUS FROM V$STANDBY_LOG;

-- Start Redo Apply

RECOVER MANAGED STANDBY DATABASE PARALLEL 8;

alter database recover managed standby database;

ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;

ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;

ALTER DATABASE RECOVER MANAGED STANDBY DATABASE USING ARCHIVED LOGFILE;

ALTER DATABASE RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE;

--Canceling a Time Delay

ALTER DATABASE RECOVER MANAGED STANDBY DATABASE NODELAY;

ALTER DATABASE START LOGICAL STANDBY APPLY NODELAY;

--Stop Redo Apply

ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;

ALTER DATABASE RECOVER MANAGED STANDBY FINISH;

--Transition to a Logical Standby Database

ALTER DATABASE RECOVER TO LOGICAL STANDBY db_name;

ALTER DATABASE RECOVER TO LOGICAL STANDBY KEEP IDENTITY; →rolling upgrade

--Transition to a PHYSICAL Standby Database

ALTER DATABASE CONVERT TO PHYSICAL STANDBY;

--start , stop LOGICAL STANDBY APPLY

ALTER DATABASE START LOGICAL STANDBY APPLY IMMEDIATE;

ALTER DATABASE STOP LOGICAL STANDBY APPLY;

--switchover

ALTER DATABASE <PREPARE | COMMIT> TO SWITCHOVER TO PHYSICAL STANDBY [WITH SESSION SHUTDOWN];

ALTER DATABASE SWITCHOVER TO usefdb VERIFY;

ALTER DATABASE SWITCHOVER TO usefdb FORCE;

ALTER DATABASE COMMIT TO SWITCHOVER TO LOGICAL STANDBY;

ALTER DATABASE COMMIT TO SWITCHOVER TO LOGICAL PRIMARY;

ALTER DATABASE PREPARE TO SWITCHOVER TO PRIMARY;

SELECT SWITCHOVER_STATUS FROM V$DATABASE;

-- FAILOVER

ALTER DATABASE FAILOVER TO usefdb;

ALTER DATABASE FAILOVER TO usefdb FORCE;

--REGISTER LOGFILE

vahidusefzadeh@gmail.com

ALTER DATABASE REGISTER [OR REPLACE] [PHYSICAL] LOGFILE <file_path_and_name>
[FOR <logminer_session_name>];

ALTER DATABASE REGISTER PHYSICAL LOGFILE '/oracle/dbs/hq_sat_12.log';

ALTER DATABASE REGISTER LOGICAL LOGFILE '/disk1/oracle/dbs/log-1292880008_7.arc';

--Activate new primary database

ALTER DATABASE ACTIVATE LOGICAL STANDBY DATABASE FINISH APPLY;

--CREATE DATAFILE

ALTER DATABASE CREATE DATAFILE [filename | filenumber] - AS [NEW | new_filename];

--RENAME DATAFILE

ALTER DATABASE RENAME FILE '/disk1/oracle/oradata/payroll/tbs_4.dbf' TO '/oracle/oradata/tbs_x.dbf';

-- CONVERT TO SNAPSHOT STANDBY

ALTER DATABASE CONVERT TO <PHYSICAL | SNAPSHOT> STANDBY;

ALTER DATABASE CONVERT TO SNAPSHOT STANDBY;

-- Unsupported Operations

ALTER DATABASE STOP LOGICAL STANDBY APPLY;

EXEC DBMS_LOGSTDBY.APPLY_SET('RECORD_UNSUPPORTED_OPERATIONS', 'TRUE');

ALTER DATABASE START LOGICAL STANDBY APPLY IMMEDIATE;

-- SKIP FAILED TRANSACTION

alter database start logical standby apply skip failed transaction'

--DG_CONFIG

ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(usefdb, usefstb)';

--At failover send unsent redo to the standby

ALTER SYSTEM FLUSH REDO SQL;

---

vahidusefzadeh@gmail.com

--Protection Mode

ALTER DATABASE SET STANDBY DATABASE TO MAXIMIZE {AVAILABILITY | PERFORMANCE | PROTECTION};

SELECT PROTECTION_MODE FROM V$DATABASE;

---Maximum Performance

log_archive_dest_2='service= usefdb _standby ARCH NOAFFIRM'

OR

log_archive_dest_2='service=usefdb_standby LGWR ASYNC NOAFFIRM'

SQL> ALTER DATABASE SET STANDBY DATABASE TO MAXIMIZE PERFORMANCE ;

---Maximum Protection

log_archive_dest_2='service= usefdb_standby LGWR SYNC AFFIRM'

SQL> ALTER DATABASE SET STANDBY DATABASE TO MAXIMIZE PROTECTION ;

---Maximum Availability

log_archive_dest_2='service= usefdb_standby LGWR SYNC AFFIRM'

SQL> ALTER DATABASE SET STANDBY DATABASE TO MAXIMIZE AVAILABILITY ;

-- Active Database Duplication

DUPLICATE TARGET DATABASE

FOR STANDBY

FROM ACTIVE DATABASE DORECOVER SPFILE

SET "db_unique_name"="foou" COMMENT ''Is a duplicate''

SET LOG_ARCHIVE_DEST_2="service=inst3 ASYNC REGISTER

VALID_FOR=(online_logfile,primary_role)"

SET FAL_SERVER="inst1" COMMENT "Is primary"

NOFILENAMECHECK;

--Backup-Based Duplication

vahidusefzadeh@gmail.com

```
DUPLICATE TARGET DATABASE
 FOR STANDBY  DORECOVER
 SPFILE
 SET "db_unique_name"="foou" COMMENT ''Is a duplicate''
 SET LOG_ARCHIVE_DEST_2="service=inst3 ASYNC REGISTER
 VALID_FOR=(online_logfile,primary_role)"
 SET FAL_SERVER="inst1" COMMENT "Is primary"
 NOFILENAMECHECK;
```

-- Role Swap On Physical Standby

```
srvctl modify database -d <physStandby> -s open -r primary
srvctl modify database -d TSTC -s open -r primary
```

--Role Swap On Primary

```
srvctl modify database -d <Primary> -s mount -r physical_standby
srvctl modify database -d TSTA -s open -r physical_standby
```

--Data Guard Broker

--Start Data Guard Manager

```
$ DGMGRL
```

--Add a standby database to the broker configuration

```
ADD DATABASE <database_name> [as connect identifier is <connect-identifier>];
```

--Connect to the database via the broker

```
DGMGRL> add database proda;
CONNECT username[@connect-identifier]
```

www.usefzadeh.com

DGMGRL> connect sys/syspwd

Connected.

Error:

ORA-16525: the Data Guard broker is not yet available

ORA-06512: at SYS:DBMS_DRS", line 124

ORA-06512: at line 1

SQL> ALTER SYSTEM SET dg_broker_start = TRUE;

System altered.

DGMGRL> connect sys/syspwd

Connected.

DGMGRL> exit

SQL> alter system set DG_BROKER_START = FALSE;

--Convert or revert a standby between physical and snapshot

CONVERT <db_unique_name> TO <SNAPSHOT | PHYSICAL> STANDBY;

DGMGRL> CONVERT DATABASE prodb TO snapshot standby;

--Create a broker configuration

CREATE CONFIGURATION <configuration-name> AS

PRIMARY DATABASE IS <database-name>

CONNECT IDENTIFIER IS <connect-identifier>;

DGMGRL> CREATE CONFIGURATION 'dg_prod_cfg' AS

PRIMARY DATABASE IS 'proda'

CONNECT IDENTIFIER IS proda.mlib.com;

--Disable a configuration

DISABLE CONFIGURATION;

DGMGRL> DISABLE CONFIGURATION;

--Disable a database

DISABLE DATABASE <database-name>;

DGMGRL> DISABLE DATABASE prodb;

--Disable Fast-Start Failover

DISABLE FAST_START FAILOVER [FORCE];

DGMGRL> DISABLE FAST_START FAILOVER FORCE;

--Disable Fast-Start Failover Condition

DISABLE FAST_START FAILOVER CONDITION <value>;

DGMGRL> DISABLE FAST_START FAILOVER CONDITION "corrupted controlfile";

--Edit a configuration property

EDIT CONFIGURATION SET PROPERTY <property_name> = <value>;

DGMGRL> EDIT CONFIGURATION SET PROPERTY FastStartFailoverThreshold=60;

--Edit configuration (protection mode)

DGMGRL> EDIT DATABASE 'site1_edrsr8p1' SET PROPERTY 'LogXptMode'='SYNC';

EDIT CONFIGURATION SET PROTECTION MODE AS <MAXPROTECTION | MAXAVAILABILITY | MAXPERFORMANCE>;

DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MAXAVAILABILITY;

--Edit database property

EDIT DATABASE <database-name> SET PROPERTY=<value>;

DGMGRL> EDIT DATABASE prodb SET PROPERTY='ArchiveLagTarget'=900;

--Edit database property

EDIT DATABASE <database-name> RENAME TO =<new-database-name>;

DGMGRL> DISABLE DATABASE prodb;

DGMGRL> EDIT DATABASE prodb RENAME TO='prodc';

DGMGRL> ENABLE DATABASE prodc;

--Edit database state

vahidusefzadeh@gmail.com

EDIT DATABASE <database-name> SET STATE=<database state>

[WITH APPLY INSTANCE=<instance-name>];

Database State

APPLY-OFF (physical or logical standby database only)

APPLY-ON (physical or logical standby database only)

TRANSPORT-OFF (primary database only)

TRANSPORT-ON (primary database only)

DGMGRL> EDIT DATABASE prodb SET STATE=apply-off;

--Edit Instance

EDIT INSTANCE <instance-name> [ON DATABASE <database-name>]

SET auto pfile=<initialization-file | off>;

DGMGRL> EDIT INSTANCE 'prodb' ON DATABASE 'prodb' SET AUTO PFILE='initprodb.ora';

DGMGRL> EDIT INSTANCE 'prodb' ON DATABASE 'prodb'

SET PROPERTY 'StandbyArchiveLocation'='/u03/prodb/arch/';

--Enable Configuration

ENABLE CONFIGURATION;

DGMGRL> ENABLE CONFIGURATION;

--Enable Database

ENABLE DATABASE <database-name>;

DGMGRL> ENABLE DATABASE prodb;

--Enable Fast_Start Failover

ENABLE FAST_START FAILOVER;

DGMGRL> ENABLE FAST_START FAILOVER;

--Enable Fast_Start Failover Condition

ENABLE FAST_START FAILOVER CONDITION <value>;

vahidusefzadeh@gmail.com

Fast_Start Failover Conditions

Corrupted Controlfile Corrupted controlfile

Corrupted Dictionary Dictionary corruption of a critical database object

Datafile Offline Data file offline due to a write error

Inaccessible Logfile LGWR is unable to write to any member of a log group due to an I/O error

Stuck Archiver Archiver is unable to archive a redo log because device is full or unavailable

DGMGRL> ENABLE FAST_START FAILOVER CONDITION "Corrupted Dictionary";

--Exit the Data Guard Manager program

EXIT;

DGMGRL> EXIT;

--Change a standby database to be the primary database

FAILOVER TO <database-name> [IMMEDIATE];

DGMGRL>FAILOVER TO prodb IMMEDIATE;

--Display description and syntax for a command

HELP;

DGMGRL> HELP;

DGMGRL> HELP SHOW;

--Exit the program

QUIT;

DGMGRL> QUIT;

--Change a disabled database into a viable standby database

REINSTATE DATABASE <database_name>;

DGMGRL>REINSTATE DATABASE prodb;

--Comment (remark) to be ignored by DGMGRL

rem <remark_string>;

DGMGRL>REM 'this is a comment';

--Removes config info, including all DB profiles and terminates management of all of associated databases

REMOVE CONFIGURATION [PRESERVE DESTINATIONS];

DGMGRL> SHOW CONFIGURATION;

DGMGRL> REMOVE CONFIGURATION PRESERVE DESTINATIONS;

DGMGRL> SHOW CONFIGURATION;

--Removes config info, including all DB profiles and terminates management of all of associated databases

REMOVE DATABASE <database_name> [PRESERVE DESTINATIONS];

DGMGRL> REMOVE DATABASE proda PRESERVE DESTINATIONS;

--Removes the specified standby database's profile from the broker configuration and terminates broker management of the standby database

REMOVE INSTANCE <instance-name> [ON DATABASE <database-name>];

DGMGRL> SHOW CONFIGURATION;

DGMGRL> REMOVE INSTANCE proda ON DATABASE proda;

--Display information about a configuration, database, and/or instance

DGMGRL> SHOW CONFIGURATION;

--Display information about a database

SHOW CONFIGURATION [VERBOSE | <property_name>];

DGMGRL> SHOW CONFIGURATION VERBOSE;

SHOW DATABASE [VERBOSE] <database-name> [<property-name>];

DGMGRL> SHOW DATABASE VERBOSE;

--Display information about a fast_start failover configuration

SHOW FAST_START FAILOVER;

DGMGRL> SHOW FAST_START FAILOVER;

--Display information about an instance

vahidusefzadeh@gmail.com

SHOW INSTANCE [VERBOSE] <instance-name> [ON DATABASE <database-name>];

DGMGRL> SHOW INSTANCE VERBOSE;

--Shutdown a currently running Oracle instance

SHUTDOWN [<ABORT | IMMEDIATE | NORMAL>];

DGMGRL> SHUTDOWN IMMEDIATE;

--Start Fast-Start Failover observer

START OBSERVER [FILE=observer_configuration_filename];

DGMGRL> START OBSERVER '/u01/oracle/admin/orabase/dgbroker/orabase_dgcfg';

--Start an Oracle database instance

STARTUP [FORCE] [RESTRICT] [PFILE=<filename>]

[<MOUNT | NOMOUNT | OPEN <READ ONLY | READ WRITE>>];

DGMGRL> STARTUP;

--Stop the Fast_Start Failover Observer

STOP OBSERVER;

DGMGRL> STOP OBSERVER;

--Switch roles between the primary database and a standby database

SWITCHOVER TO <database-name>;

DGMGRL> SWITCHOVER TO prodc;

--useful query:

```
select THREAD#,max(SEQUENCE#) from gv$archived_log where applied='YES'and RESETLOGS_ID=865773832
group by thread#;
select SEQUENCE#,THREAD#,inst_id,STATUS,BLOCK# from gv$managed_standby where lower(status) like '%wait%'
or lower(status) like '%ap%';
```