RMAN Active Cloning In Oracle 12C

RMAN active is one of cloning method used to clone a database on target host without taking backup of the source database.  Make sure that source database is in archivelog mode. Below are the steps.

***NOTE****:*

*In 11g, RMAN active cloning is done during image copy( which includes unused blocks) . It means if a datafile is of 31 GB, but only 10GB databas is present, then it will copy the whole 31GB over network.   And it uses the PUSH method , which increases the load on SOURCE host.*

*Where as in 12C, RMAN cloning is done using BACKUPSET, i.e instead of 31G, only 10GB will be copied( that is the used data), Which makes it more efficient. It used PULL method,*

EXAMPLE:

SOURCE DB – oratest

TARGET DB – oratest2 ( Also called auxiliary instance )

***NOTE – If your target database already exists, then drop the database before starting the cloning.***

**1. Add the tns entry of the both database in  tnsnames.ora file of target host :**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26 | -- source db tns :    ORATEST =  (DESCRIPTION =  (ADDRESS = (PROTOCOL = TCP)(HOST = npvslxdc64.npower.com)(PORT = 1529))  (CONNECT\_DATA =  (SERVER = DEDICATED)  (SERVICE\_NAME = oratest)  )  )    --Target db tns :      ORATEST2 =  (DESCRIPTION =  (ADDRESS = (PROTOCOL = TCP)(HOST = npvslxdc63.npower.com)(PORT = 1529))  (CONNECT\_DATA =  (SERVER = DEDICATED)  (SERVICE\_NAME = oratest2)  )    ) |

**2. Create a listener for the target db ( with static registration)**

The relevant one is that - if you want to remotely startup the database, you need to have a statically registered connect string, otherwise - since the database isn't up - it wouldn't be registered. 

Use dynamic registration for all application connects - all of them. Use static only if and when you need to remotely start the database over the network.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21 | LISTENER\_ORATEST2 =  (DESCRIPTION\_LIST =  (DESCRIPTION =  (ADDRESS = (PROTOCOL = TCP)(HOST = npvslxdc63.npower.com)(PORT = 1529))  )  )    SID\_LIST\_LISTENER\_ORATEST2 =  (SID\_LIST =  (SID\_DESC =  (GLOBAL\_DBNAME = ORATEST2)  (ORACLE\_HOME = /oracle/RMANCAT/12.2.0.1)  (SID\_NAME = ORATEST2 )  ))  -- START THE LISTENER    lsnrctl start LISTENER\_ORATEST2 |

**3. Copy the pfile from source host :**

Copy the pfile from source host to target host and modify the parameters like control\_files,diagnostic\_dest,audit\_dump .  
Apart from that add below two (mandatory) parameters in the the pfile of target db pfile.

\*.db\_file\_name\_convert = (“< source db db file location> “,”< target db db file location>”)  
\*.log\_file\_name\_convert= (“< sourcec db redo log location>”,”<target db redo log location”)

it will look as :

oratest2.\_\_inmemory\_ext\_rwarea=0

oratest2.\_\_java\_pool\_size=4194304

oratest2.\_\_large\_pool\_size=71303168

oratest2.\_\_oracle\_base='/oracle/RMANCAT'#ORACLE\_BASE set from environment

oratest2.\_\_pga\_aggregate\_target=524288000

oratest2.\_\_sga\_target=1048576000

oratest2.\_\_shared\_io\_pool\_size=29360128

oratest2.\_\_shared\_pool\_size=369098752

oratest2.\_\_streams\_pool\_size=0

\*.audit\_file\_dest='/oracle/RMANCAT/admin/oratest2/adump'

\*.audit\_trail='db'

\*.compatible='12.2.0'

\*.control\_files='/oracle/RMANCAT/data01/CDB\_TEST/control01.ctl','/oracle/RMANCAT/data01/CDB\_TEST/control02.ctl'

\*.db\_block\_size=8192

\*.db\_name='oratest2'

\*.diagnostic\_dest='/oracle/RMANCAT'

\*.dispatchers='(PROTOCOL=TCP) (SERVICE=oratestXDB)'

\*.enable\_pluggable\_database=true

\*.local\_listener='LISTENER\_ORATEST2'

\*.nls\_language='ENGLISH'

\*.nls\_territory='UNITED KINGDOM'

\*.open\_cursors=300

\*.pga\_aggregate\_target=500m

\*.processes=640

\*.remote\_login\_passwordfile='EXCLUSIVE'

\*.service\_names='ORATEST2,PDB1'

\*.sga\_target=1000m

\*.undo\_tablespace='UNDOTBS1'

\*.db\_file\_name\_convert=('/oracle/RMANSTD/data01/CDB\_TEST','/oracle/RMANCAT/data01/CDB\_TEST','/oracle/RMANSTD/data01/CDB\_TEST/PDB1','/oracle/RMANCAT/data01/CDB\_TEST', '/oracle/RMANSTD/data01/CDB\_TEST/PDBCLONE','/oracle/RMANCAT/data01/CDB\_cd TEST/PDBCLONE','/oracle/RMANSTD/data01/CDB\_TEST/PDBSEED','/oracle/RMANCAT/data01/CDB\_TEST/PDBSEED')

\*.log\_file\_name\_convert=('/oracle/RMANSTD/data01/CDB\_TEST','/oracle/RMANCAT/data01/CDB\_TEST')

**4. Create password file on both source and target db ( keep same password )**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | -- SOURCE DB ( PRODDB)    cd $ORACLE\_HOME/dbs  orapwd file=orapw$ORACLE\_SID password=notsys force=y    -- TARGET DB ( TESTDB)    cd $ORACLE\_HOME/dbs  orapwd file=orapw$ORACLE\_SID password=oracle force=y |

**5. Start the target db/auxiliary instance (oratest2) in nomount state:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | export ORACLE\_SID=oratest2  SQL> create spfile from pfile ;  SQL> Startup nomount |

**6. Check the connection to both source and target as below**

run the below command from TEST DB :

rman target sys/oracle@< source db\_tns\_name>  auxiliary sys/oracle@<target db\_tns\_name>

i.e  
rman target sys/notsys@oratest auxiliary sys/notsys@oratest2

***If you are getting any error while running this command, then fix the same, before proceeding further.***

**7 . Start the cloning:**

Now run the below rman script from target db host:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | rman target sys/notsys@oratest auxiliary sys/oracle@oratest2    run  {  allocate channel src1 type disk;  allocate channel src2 type disk;  allocate auxiliary channel aux1 type disk;  allocate auxiliary channel aux2 type disk;  allocate auxiliary channel aux3 type disk;  allocate auxiliary channel aux4 type disk;  duplicate target database to oratest2 from active database USING  BACKUPSET ;  } |

Once this script completed, it will open the target db ( oratest2) in resetlog mode.  With this cloning completes.

*NOTE: If your oracle version is 11g, then use the below rman script.*

*run  
{  
allocate channel src1 type disk;  
allocate channel src2 type disk;  
allocate auxiliary channel aux1 type disk;  
allocate auxiliary channel aux2 type disk;  
allocate auxiliary channel aux3 type disk;  
allocate auxiliary channel aux4 type disk;  
duplicate target database to TESTDB from active database;  
}*