Database refresh

# Refresh :

=======

• refresh is nothing but a copy of data from one database server to another database server.

• It may be a partial copy or full database copy.

# Types of refresh :

==============

1. table refresh

2. Schema refresh

3. Full database refresh

# Traditional exp imp or datapump :

==========================

1. table

2. Schema

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Table refresh:

============

• Take the table backup

• Cope dumpfile to target server

Scp table\_t1.dmp oracle@target ip address:location.

# imp in target server:

================

• imp system/system file=dumpfile.dmp log=imp\_t1.log fromuser=u1 touser=u1 tables=t1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. schema refresh:

=============

• Take the schema backup.

• Copy dumpfile to target server.

Scp dumpfile.dmp ::::::::::::::::::::

# imp in target server:

===============

• imp system/system directory=expdp\_mouli file=dumpfile.dmp log=imp\_schema.log fromuser=u1 touser=u2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Table\_exists\_action :

================

1. Replace

2. Append

3. Truncate

• if the importing table is present in the server then import will be skip .

1. replace:

=======

• It will drop the existing table and recreate with source data.

• table\_exists\_action=replace

2. Append:

=======

• It won’t drop the table .it will add source data to the existing table data

• In this if the table has primary key it will fail.

• table\_exists\_action=append

3. Truncate:

========

• it will truncate the table and import the source data.

• table\_exists\_action=truncate

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# if source schema and target schema is different:

==============================

• u1:u2

• In target schema no tables should be present.

• take expdp backup

• Copy dumpfile from source to target

• Scp====================

• impdp system/system dumpfile=schema\_u1.dmp logfile=impdp\_u2.log directory= impdp\_mouli remap\_schema=u1:u2 schemas=u1

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# full database refresh:

=================

• copy pfile from source to target in every full db refresh.

• source and target must be same version.

• Source and target must have same operating system.

• source db name and target db name is different

# conventional cold refresh :

====================

Source db:

~~~~~~~~

1. check C R D files

2. shut down database

3. Take cold backup /copy to target server

4. Copy pfile to target server

5. Put db in mount state take controlfile trace

Target db:

~~~~~~~

1. create required directories ( as in source db)

2. Set env (copy from source) change Sid name.

3. Edit the pfile ( mv inithyd.ora inithydtst.ora)

• In pfile change db names

4. Put db in nomount state

5. Re create the controlfile ( trace file script)

6. Open the database with resetlogs

• Alter database open resetlogs;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# conventional hot backup:

====================

• copy pfile from source to target in every full db refresh.

source :

~~~~~~

• directory structure /prod/hyd/oradata

1. db must be enable with archivelog mode

2. Check the max sequence

3. Put the db in begin backup mode

4. Take backup and copy to target server

5. End the begin backup mode

6. do some log switches

7. Check max sequence

8. Once again check the archives count and copy to the backup location ki

9. Copy pfile to target server

Target:

~~~~~

• directory structure/prod/hydtst/oradata

1. Create required directories

2. Set env file

3. Edit pfile

4. Put db in nomount state

5. Recreate the controlfile with reserlogs option.

6. Recover database using backup controlfile until cancel;

7. Alter database open resetlogs;

===================================

# RMAN database refresh:

====================

1. Rman cold backup

2. Rman hot backup

3. Incremental backup

4. Auxiliary duplicate method (backup is required)

5. active duplicate method (backup is not required)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Rman cold database refresh:

========================

Source side :

~~~~~~~~~

1. Rman cold backup

2. Controlfile backup

3. Copy backup to target server

4. Copy pfile

Target server:

~~~~~~~~~~

1. create required directories

2. Set env

3. Edit pfile

Initially db name is same as prod db name (hyd) change remaining names

4. put the db in nomount state

5. Restore controlfile from backup /connect to Rman / ( RMAN >restore controlfile from ‘location , file name’;)

6. Put db in mount ( RMAN> alter database mount.)

7. Register backup in target database ( catalog start with ‘backup file location’;)

8. Restore database

• When the db datafiles locations are different we have to use this script.

~~~~~~~~~~~~~~~~~~~~~~~~~~

run

{

Set newname for datafile 1 to ‘/prod/hydtst/oradata/system01.dbf’;

Set newname for datafile 2 to ‘/prod/hydtst/oradata/sysaux01.dbf’;

Set newname for datafile 3 to ‘/prod/hydtst/oradata/undotbs01.dbf’;

Set newname for datafile 4 to ‘/prod/hydtst/oradata/users01.dbf’;

restore database;

}

9. Update datafiles location in controlfile RMAN>switch database to copy;

10. Exit from rman

11. rename redo log files ( alter database rename file ‘/prod/hyd/redo’ to ‘/prod/hydtst/redo’; (all)

12. open the database with resetlogs ( alter database open resetlogs;)

13. Add temp file( alter tablespace temp add tempfile ‘/prod/hydtst/oradata/temp\_01.dbf’ size 100m; ( drop old temp file)

14. Change db name using NID tool

15. To use NID tool

• tempfile

• Password file

• Put the db in mount state

NID help= y

• nid target= system/system dbname=hydtst SETNAME=y

16. change db name in pfile.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Auxiliary duplicate database refresh:

=============================

Source:

======

1. Take Rman backup level 0,1, archive all.

2. Copy backup to target server

3. Copy pfile to target server

Target:

=====

1. create required directories

2. Set env

3. Edit pfile

db\_name= ramtst

db\_file\_name\_convert=‘/prod/ram/oradata/‘,’/prod/ramtst/oradata/‘

Log\_file\_name\_convert= ‘/prod/ram/oradata/‘,’/prod/ramtst/oradata/‘

4. put db in nomount state

5. Connect to rman

Rman auxiliary /

Rman>duplicate target database to ramtst backup location ‘/prod/ramtst/backup/‘ nofilenamecheck;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Active duplicate database refresh:

===========================

Source:

======

1. tnsnames

2. Password file

3. copy pfile to target

4. connect to Rman

Rman target / auxiliary=system/system@tns

Rman>duplicate target database to ramtst backup location ‘/prod/ramtst/backup/‘ nofilenamecheck;

Target:

=====

1. Create required directories

2. Set env file

3. Edit pfile

db\_name= ramtst

db\_file\_name\_convert=‘/prod/ram/oradata/‘,’/prod/ramtst/oradata/‘

Log\_file\_name\_convert= ‘/prod/ram/oradata/‘,’/prod/ramtst/oradata/‘

4. put db in nomount stat

5. Listener

6. Password file copy from source

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_