

The background of the slide is a photograph of a large audience at an Oracle user conference. The scene is dimly lit with vibrant red and blue stage lighting. In the upper right, a large screen displays the Oracle logo and some technical content. The audience is seen from behind, looking towards the stage area.

Oracle 12c Snapshot Standby Database szerepe és konfigurálása

Gecseg Gyula

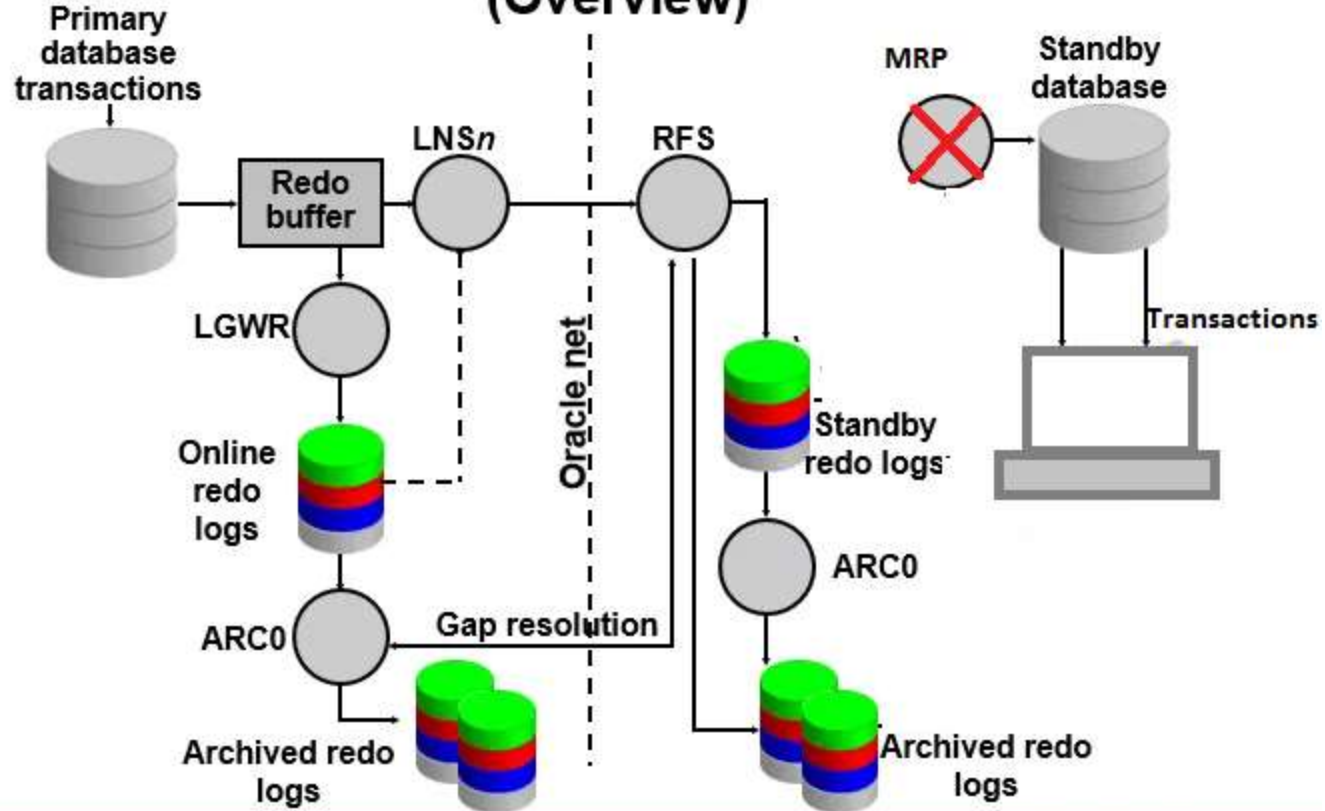
Oracle DBA

Oracle user konferencia

The HOUG logo consists of a grid of white dots arranged in a roughly rectangular shape, with the word "HOUG" in a bold, white, sans-serif font positioned directly below the grid.

HOUG

Oracle Data Guard: Architecture (Overview)



Bevezetés

Az Oracle a 11g verzióban vezette be a Snapshot Standby feature-t, lényege hogy megengedi a read-write műveleteket a Standby adatbázison.

Nagy előnye, hogy minden tesztet képesek vagyunk elvégezni a Produktív környezet adataival anélkül hogy az éles Produktív környezet adatait veszélyeztetnénk.

- Egy Snapshot Standby adatbázis megkapja az archive logokat a Primary adatbázisból, de nem alkalmazza őket
- Az összes rendelkezésre álló redo log akkor lesz alkalmazva, amikor a Snapshot Standby vissza lesz konvertálva Physical standby adatbázissá

A tesztek elvégzése után az adatbázist visszaállítjuk a Flashback Database funkcióval, majd visszakonvertáljuk Physical Standby database állapotba.

Flashback Database előnyei Data Guard környezetben

Az Oracle Flashback adatbázis-technológiájával gyorsan hozhatja az adatbázist egy korábbi időpontba azáltal, hogy visszavonja az azóta történt változásokat. Az adatbázis visszajátszást (replay) képesek vagyunk futtatni többször is a tesztrendszeren a produktív rendszer pontos időzítésével, tranzakciós jellemzőivel és terhelésével.

Előnyei:

Lekérdezések hangolása

Az alkalmazásgazdák meg tudják nyitni egy korábbi időpontra és különböző lekérdezéseket tudnak futtatni rajta.

Ideális valós alkalmazás teszteléshez

Nincs szükség extra storage területre

katasztrófavédelem (Disaster recovery) fenntartása tesztelés közben

Real Application Testing (RAC) környezethez köthető előnyök Database Rplay esetén

- Az adatbázis és a rendszergazdák tesztelhetik az adatbázis frissítéseket, javításokat, paraméterváltozásokat.
- Konfiguráció változtatásokat
 - Konverzió single példányból RAC - ASM-re
- Storage, hálózat változtatások (interconnect changes)
- Operációs rendszer, hardver migrációk, javítások, frissítések, paraméterváltoztatások
- SQL Performance Analyzer használata (a *Real Application Testing* része)

Flashback Database előnyei Data Guard környezetben

SQL Performance Analyzer (Real Application Testing) használata

Az SQL Performance Analyzer-t a egy Snapshot Standby adatbázis példányon futtattjuk, a Snapshot Standby adatbázisnak előfeltétele a Flashback Database.

- Adatbázis frissítés, javítások, inicializálási paraméterek változása
- Az operációs rendszer, a hardver vagy az adatbázis konfigurációs változásai
- A séma változásai, mint például új indexek, partíciók vagy materializált nézetek
- Optimalizáló statisztikák összegyűjtése
- SQL hangolási műveletek, például SQL profilok létrehozása
- Adatbázis konszolidációs tesztelés az Oracle Multitenant vagy séma konszolidáció segítségével

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary database

1. Engedélyezzük az archive módot:

```
SQL>  
SQL> select log_mode from v$database;
```

```
LOG_MODE  
-----
```

```
ARCHIVELOG
```

```
SQL> █
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary database

2. Engedélyezzük a force logging módot:

```
SQL> select force_logging from v$database;
```

```
FORCE_LOGGING
```

```
-----  
YES
```

```
SQL> █
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary database

3. Konfiguráljuk a „Standby-Redo log” fajt, ezeket fogja használni „switchover vagy failover” esetén.

```
SQL> select GROUP#, BYTES/1024/1024, STATUS, FIRST_TIME, NEXT_TIME from v$standby_log;
```

```
GROUP# BYTES/1024/1024 STATUS  FIRST_TIM NEXT_TIME
-----
4      55 UNASSIG NED
5      55 UNASSIG NED
6      55 UNASSIG NED
7      55 UNASSIG NED
```


Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary database

4. Állítsuk be a Primary adatbázis inicializációs paramétereit.

```
alter system set log_archive_dest_1='LOCATION=use db_recovery_file_dest valid_for=(ALL_ROLES,ALL_LOGFILES) db_unique_name=TEST' scope=both sid='*';
alter system set log_archive_dest_2='service="TEST_ST",ASYNC NOAFFIRM delay=0 optional compression=disable max_failure=0 max_connections=1 reopen=300 db_unique_name="TEST_ST" net_timeout=30',
'valid_for=(online_logfile,all_roles)' scope=both sid='*';
alter system set log_archive_dest_state_2=defer scope=both sid='*';
alter system set standby_file_management=auto scope=both sid='*';
alter system set log_archive_config='dg_config=(TEST,TEST_ST)' scope=both sid='*';
alter system set dg_broker_start=TRUE scope=both sid='*';
alter system set fal_client='TEST_ST' scope=both sid='*'; #to_primary
alter system set fal_server='TEST' scope=both sid='*'; #to_standby
alter system set DB_FILE_NAME_CONVERT='/oracle/oradata/TEST', '/oracle/oradata/TEST_ST' scope=spfile sid='*';
alter system set log_file_name_convert='/oracle/oradata/TEST', '/oracle/oradata/TEST_ST' scope=spfile sid='*';
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary database

5. Konfiguráljuk a listener.ora és a tnsnames.ora fájlt.

```
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
  )
)

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = TEST)
      (ORACLE_HOME = /oracle/12.2.0.2/12.1.0)
      (SID_NAME = TEST)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TEST_ST)
      (ORACLE_HOME = /oracle/12.2.0.2/12.1.0)
      (SID_NAME = TEST_ST)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TEST_ST1)
      (ORACLE_HOME = /oracle/12.2.0.2/12.1.0)
      (SID_NAME = TEST_ST1)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TEST_DGMGRL)
      (ORACLE_HOME = /oracle/12.2.0.2/12.1.0)
      (SID_NAME = TEST)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TEST_ST_DGMGRL)
      (ORACLE_HOME = /oracle/12.2.0.2/12.1.0)
      (SID_NAME = TEST_ST)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TEST_ST1_DGMGRL)
      (ORACLE_HOME = /oracle/12.2.0.2/12.1.0)
      (SID_NAME = TEST_ST1_DGMGRL)
    )
  )
)
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary database

5. Konfiguráljuk a listener.ora és a tnsnames.ora fájlt.

```
LISTENER_TEST =
  (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))

TEST_ST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = TEST_ST)
    )
  )

TEST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = TEST)
    )
  )

TEST_ST1 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = localhost)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = TEST_ST1)
    )
  )
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

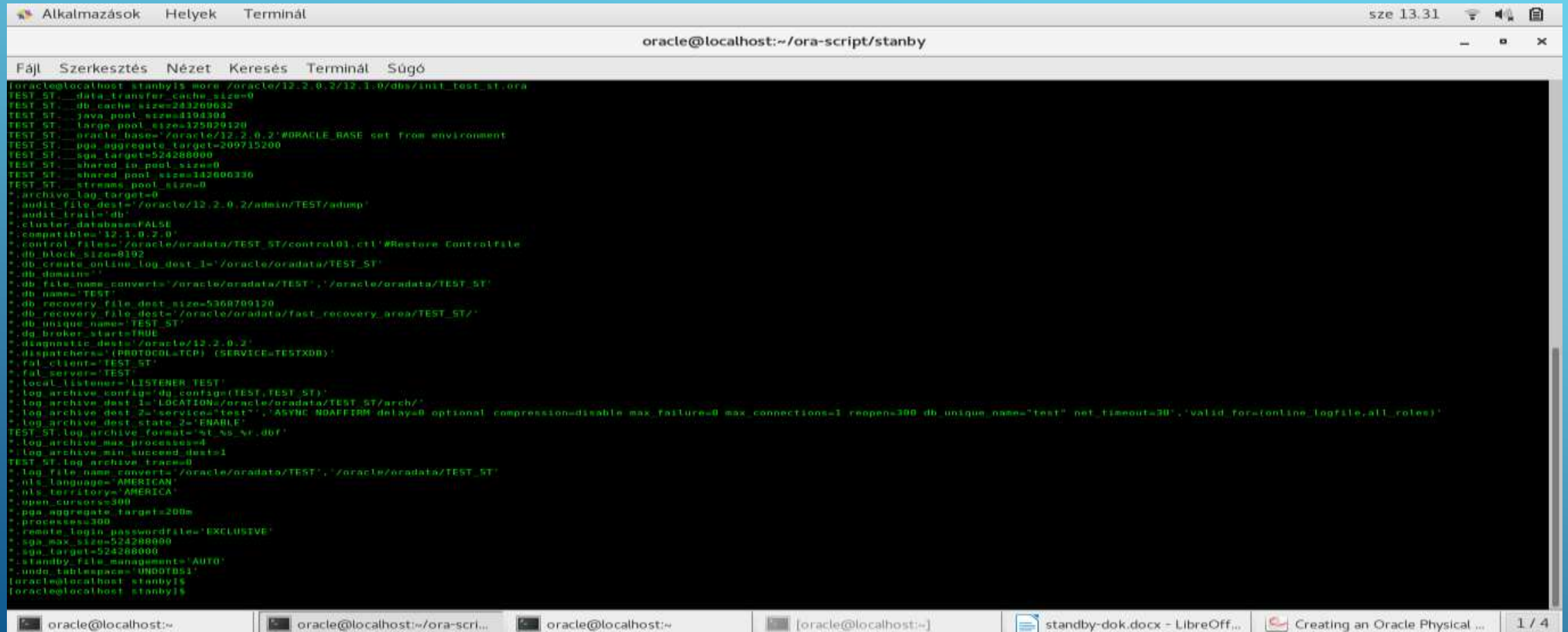
Prepare the Standby database

6. Hozzuk létre a standby adatbázis számára szükséges könyvtárakat.

```
#mkdir $ORACLE_BASE/admin/TEST_ST  
#mkdir $ORACLE_BASE/admin/TEST_ST/adump  
#mkdir $ORACLE_BASE/admin/TEST_ST/cdump  
#mkdir $ORACLE_BASE/admin/TEST_ST1  
#mkdir $ORACLE_BASE/admin/TEST_ST1/adump  
#mkdir $ORACLE_BASE/admin/TEST_ST1/cdump
```

Snapshot Standby Database megvalósítása egy példával: Egy Physical standby installálása és aktiválása

Prepare the Standby database 7. Hozzuk létre a szükséges init.ora fájlt.



```
oracle@localhost:~/ora-script/standby$ more /oracle/12.2.0.2/12.1.0/dbs/init_test_st.ora
TEST_ST__data_transfer_cache_size=0
TEST_ST__db_cache_size=243200632
TEST_ST__java_pool_size=104304
TEST_ST__large_pool_size=125829120
TEST_ST__oracle_base='/oracle/12.2.0.2'#ORACLE_BASE set from environment
TEST_ST__pga_aggregate_target=209715200
TEST_ST__sga_target=524288000
TEST_ST__shared_io_pool_size=0
TEST_ST__shared_pool_size=142000330
TEST_ST__streams_pool_size=0
*.archive_log_target=0
*.audit_file_dest='/oracle/12.2.0.2/admin/TEST/adump'
*.audit_trail='db'
*.cluster_databases=FALSE
*.compatible='12.1.0.2.0'
*.control_files='/oracle/oradata/TEST_ST/control01.ctl'#Restore Controlfile
*.db_block_size=8192
*.db_create_online_log_dest_1='/oracle/oradata/TEST_ST'
*.db_domain=''
*.db_file_name_convert='/oracle/oradata/TEST','/oracle/oradata/TEST_ST'
*.db_name='TEST'
*.db_recovery_file_dest_size=5368709120
*.db_recovery_file_dest='/oracle/oradata/fast_recovery_area/TEST_ST/'
*.db_unique_name='TEST_ST'
*.dg_broker_start=TRUE
*.diagnostic_dest='/oracle/12.2.0.2'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=TESTXDB)!'
*.fal_client='TEST_ST'
*.fal_server='TEST'
*.local_listener='LISTENER_TEST'
*.log_archive_config='dg_config(TEST,TEST_ST)'
*.log_archive_dest_1='LOCATION=/oracle/oradata/TEST_ST/arch/'
*.log_archive_dest_2='service=test',ASYNC NOAFFIRM delay=0 optional_compression=disable max_failures=0 max_connections=1 reopen=300 db_unique_name=test net_timeout=30,'valid_for=(online_logfile,all_roles)'
*.log_archive_dest_state_2='ENABLE'
TEST_ST.log_archive_format='%t_%s_%r.dbf'
*.log_archive_max_processes=4
*.log_archive_min_succeed_dest=1
TEST_ST.log_archive_traces=0
*.log_file_name_convert='/oracle/oradata/TEST','/oracle/oradata/TEST_ST'
*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.open_cursors=300
*.pga_aggregate_target=200m
*.processes=300
*.remote_login_password_file='EXCLUSIVE'
*.sga_max_size=524288000
*.sga_target=524288000
*.standby_file_management='AUTO'
*.undo_tablespace='UNDOTBS1'
oracle@localhost: standby$
oracle@localhost: standby$
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

8. A standby adatbázis számára hozzunk létre egy password fájlt.

```
orapwd file=/oracle/12.2.0.2/12.1.0/dbs/orapwTEST_ST password=valami entries=10
```

9. Indítsuk el a standby instance-t nomount-tal.

```
[oracle@localhost stanby]$ sqlplus / as sysdba  
SQL*Plus: Release 12.1.0.2.0 Production on Wed Feb 27 14:14:10 2019  
Copyright (c) 1982, 2014, Oracle. All rights reserved.  
Connected to an idle instance.  
SQL> startup nomount pfile='/oracle/12.2.0.2/12.1.0/dbs/init_test_st.ora';
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

10. Kapcsolódjunk az RMAN TARGET-hez.

```
#rman target sys/valami@TEST auxiliary sys/valami@TEST_ST1
```

11. Használjuk a „DUPLICATE DATABASE” parancsot.

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

11. Használjuk a „DUPLICATE DATABASE” parancsot.

```
run {
duplicate target database for standby from active database
spfile
set 'db_unique_name'='TEST_ST'
set 'cluster_database'='false'
set 'db_create_online_log_dest_1'='/oracle/oradata/TEST_ST'
set 'log_file_name_convert'='/oracle/oradata/TEST','/oracle/oradata/TEST_ST'
set db_recovery_file_dest_size='5G'
SET SGA_MAX_SIZE '500M'
SET SGA_TARGET '500M'
set CONTROL_FILES='/oracle/oradata/TEST_ST/control01.ctl'
set dg_broker_config_file1='/oracle/12.1.0/dbs/dr1test_st.dat'
set dg_broker_config_file2='/oracle/12.1.0/dbs/dr2test_st.dat'
nofilenamecheck
;
}
```


Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

12. Állítsuk be a standby adatbázis inicializációs paramétereit.

```
[oracle@localhost stanby]$ cat test-dr-para-st.txt
alter system set log_archive_config='dg_config=(TEST,TEST_ST)' scope=both sid='*';
alter system set dg_broker_config_file1='/oracle/12.2.0.2/12.1.0/dbs/dr1TEST_ST' scope=both sid='*';
alter system set dg_broker_config_file2='/oracle/12.2.0.2/12.1.0/dbs/dr2TEST_ST' scope=both sid='*';
alter system set dg_broker_start=TRUE scope=both sid='*';
alter system set fal_client='TEST_ST' scope=both sid='*';
alter system set fal_server='TEST' scope=both sid='*';
alter system set DB_FILE_NAME_CONVERT='/oracle/oradata/TEST','/oracle/oradata/TEST_ST' scope=spfile sid='*';
alter system set log_file_name_convert='/oracle/oradata/TEST','/oracle/oradata/TEST_ST' scope=spfile sid='*';
alter system set "_query_on_physical"=false sid='*'; # A vege
alter system set remote_os_authent = FALSE scope=both sid='*';
```

Prepare the Primary database

13. Primary és Standby adatbázis státusz és szinkron ellenőrzése.

```
SQL> select name, open_mode, database_role from v$database;
```

NAME	OPEN_MODE	DATABASE_ROLE
TEST	READ WRITE	PRIMARY

```
SQL> █
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

13. Primary és Standby adatbázis státusz és szinkron ellenőrzése.

```
SQL> select status, instance_name, database_role, open_mode from v$database, v$instance;
```

```
STATUS          INSTANCE_NAME    DATABASE_ROLE    OPEN_MODE
-----
MOUNTED         TEST_ST         PHYSICAL STANDBY MOUNTED
```

```
SQL> █
```

```
Thread Last Sequence Received Last Sequence Applied Difference
```

```
-----
1 62 62 0
```

```
SQL> █
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

13. Primary és Standby adatbázis státusz és szinkron ellenőrzése.

```
SQL> select status, instance_name, database_role, open_mode from v$database, v$instance;
```

STATUS	INSTANCE_NAME	DATABASE_ROLE	OPEN_MODE
MOUNTED	TEST_ST1	PHYSICAL STANDBY	MOUNTED

```
Thread Last Sequence Received Last Sequence Applied Difference
```

1	62	62	0
---	----	----	---

```
SQL>
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

14. Indítsuk el az archive fájlok rágörgetését.

```
SQL>
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;

Database altered.
```

```
SQL>
SQL> select GROUP#, BYTES/1024/1024, STATUS, FIRST_TIME, NEXT_TIME from v$standby_log;

  GROUP# BYTES/1024/1024 STATUS      FIRST_TIM NEXT_TIME
-----
         4              55 UNASSIGNED
         5              55 ACTIVE       27-FEB-19
         6              55 UNASSIGNED
         7              55 UNASSIGNED

SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE disconnect;

Database altered.
```

```
SQL>
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary and Standby database

15. Konfiguráljuk a Data Guard Broker-t.

```
DGMGRL>  
DGMGRL> CREATE CONFIGURATION test_dg_config AS PRIMARY DATABASE IS TEST CONNECT IDENTIFIER IS TEST;
```

```
DGMGRL>  
DGMGRL> CREATE CONFIGURATION test_dg_config AS PRIMARY DATABASE IS TEST CONNECT IDENTIFIER IS TEST;
```

```
DGMGRL>  
DGMGRL> ENABLE CONFIGURATION;
```

```
DGMGRL>  
DGMGRL> SHOW CONFIGURATION;  
  
Configuration - test_dg_config  
  
Protection Mode: MaxPerformance  
Members:  
test - Primary database  
TEST_ST - Physical standby database  
TEST_ST1 - Physical standby database  
  
Fast-Start Failover: DISABLED  
  
Configuration Status:  
SUCCESS (status updated 43 seconds ago)
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Primary and Standby database

15. Konfiguráljuk a Data Guard Broker-t.

```
DGMGRL> show database "TEST_ST1"
```

```
Database - TEST_ST1
```

```
Role:                PHYSICAL STANDBY
Intended State:      APPLY-ON
Transport Lag:       0 seconds (computed 1 second ago)
Apply Lag:           0 seconds (computed 1 second ago)
Average Apply Rate: 0 Byte/s
Real Time Query:     OFF
Instance(s):
  TEST_ST1
```

```
Database Status:
SUCCESS
```

Snapshot Standby Database megvalósítása egy példával:

Egy Physical standby installálása és aktiválása

Prepare the Standby database

16. Konvertáljuk a physical standby adatbázist – snapshot standby adatbázissá

```
DGMGRL> convert database "TEST_ST1" to snapshot standby;  
Converting database "TEST_ST1" to a Snapshot Standby database, please wait...  
Database "TEST_ST1" converted successfully  
DGMGRL>
```

17. Konvertáljuk a snapshot standby adatbázist physical standby adatbázissá

```
DGMGRL> convert database "TEST_ST1" to physical standby;  
Converting database "TEST_ST1" to a Physical Standby database, please wait...  
Database "TEST_ST1" converted successfully  
DGMGRL>
```