

Création d'une logical standby database

Table des matières

1.	Configuration listeners (listener.ora)	2
2.	Créer une logical standby database	3
3.	Mise en Dataguard de la base de données primaire.....	3
4.	Création de la physical standby database	3
5.	Démarrage de l'instance TESTDG_DG02 en NOMOUNT.....	5
6.	Création de la standby database avec RMAN	5
7.	Activation de la flashback database côté physical standby database.....	5
8.	Ajout de la physical standby database dans Dataguard.....	5
9.	Préparation des tables sans clé primaire pour la conversion en logical standby	6
10.	Préparation de l'environnement pour la conversion de la standby de physique à logique ...	6
11.	Ouverture de la logical standby en RESETLOGS	6
12.	Démarrage de l'application des redologs en provenance de la primary database.....	7
13.	Enregistrement de la logical standby dans Dataguard.....	7

1. Configuration listeners (listener.ora)

Le listener doit écouter obligatoirement sur deux global_dbname particuliers :

<db_unique_name>_DGB et <db_unique_name>_DGMGRL. Ces deux noms globaux sont essentiels pour Dataguard et permettent d'accéder à l'instance sous forme service_name même si celle-ci est arrêtée.

Configuration des alias de connexion (tnsnames.ora) :

Les alias de connexion doivent être déclarés sur les services spéciaux Dataguard avec une

Serveur dataguard01

```
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
      (ADDRESS = (PROTOCOL = TCP) (HOST = dataguard01) (PORT = 1521))
    )
  )
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = TESTDG_DG01_DGB)
      (ORACLE_HOME = /oracle/ora112)
      (SID_NAME = TESTDG)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TESTDG_DG01_DGMGRL)
      (ORACLE_HOME = /oracle/ora112)
      (SID_NAME = TESTDG)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TESTDG_DG01)
      (ORACLE_HOME = /oracle/ora112)
      (SID_NAME = TESTDG)
    )
  )
DEFAULT_SERVICE_LISTENER=TESTDG_DG01_DGB
ADR_BASE_LISTENER = /oracle
```

Serveur dataguard02

```
LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
      (ADDRESS = (PROTOCOL = TCP) (HOST = dataguard02) (PORT = 1521))
    )
  )
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = TESTDG_DG02_DGB)
      (ORACLE_HOME = /oracle/ora112)
      (SID_NAME = TESTDG)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = TESTDG_DG02_DGMGRL)
      (ORACLE_HOME = /oracle/ora112)
      (SID_NAME = TESTDG)
    )
  )
```

```
(SID_DESC =  
  (GLOBAL_DBNAME = TESTDG_DG02)  
  (ORACLE_HOME = /oracle/ora112)  
  (SID_NAME = TESTDG)  
)  
)  
DEFAULT_SERVICE_LISTENER=TESTDG_DG01_DGB  
ADR_BASE_LISTENER = /oracle
```

2. Créer une logical standby database

Activation de la flashback database côté base de données primaire :

```
alter database flashback on;
```

3. Mise en Dataguard de la base de données primaire

L'enregistrement de la base de données primaire dans Dataguard se fait en créant la configuration Dataguard :

```
DGMGRL> create configuration testdg as primary database is testdg_dg01 connect  
identifier is testdg_dg01_dg;
```

Configuration "testdg" created with primary database "testdg_dg01"

```
DGMGRL> enable configuration
```

4. Création de la physical standby database

La création de la physical standby database peut être faite à l'aide de RMAN et de la base de données primaire. L'utilisation de la base de données primaire (primary database) pour créer la physical standby permet de s'affranchir de l'utilisation d'une sauvegarde.

Copie du fichier orapw<SID>

Dans une configuration Dataguard, les bases de données doivent avoir strictement le même fichier de mots de passe SYS. Ainsi le fichier créé sur la base de données primaire doit simplement être recopié d'un serveur à l'autre, dans le répertoire \$ORACLE_HOME/dbs :

```
[oracle@dataguard02 ~]$ cd $ORACLE_HOME/dbs
```

```
[oracle@dataguard02 dbs]$ scp dataguard01:/oracle/ora112/dbs/orapwTESTDG
```

Préparation du fichier init.ora pour spfile de la physical standby

Le fichier init<SID>.ora servant à la création du spfile de la standby database peut être créé directement à partir de la base de données primaire (dataguard01, instance TESTDG) :

```
create pfile='/tmp/initTESTDG.ora.forstdby' from spfile;
```

Copier le fichier vers le serveur dataguard02, l'éditer et remplacer toutes les références à TESTDG_DG01 par TESTDG_DG02. Ajouter les db_file_name_convert et log_file_name_convert nécessaires au bon positionnement des fichiers de l'instance TESTDG_DG02 sur le serveur dataguard02 :

```
TESTDG.__db_cache_size=771751936
TESTDG.__java_pool_size=16777216
TESTDG.__large_pool_size=16777216
TESTDG.__oracle_base='/oracle'#ORACLE_BASE set from environment
TESTDG.__pga_aggregate_target=369098752
TESTDG.__sga_target=1107296256
TESTDG.__shared_io_pool_size=0
TESTDG.__shared_pool_size=285212672
TESTDG.__streams_pool_size=0
*.archive_lag_target=0
*.audit_file_dest='/oracle/admin/TESTDG/adump'
*.audit_trail='db'
*.compatible='11.2.0.0.0'
*.control_files='+DATA,+FRA'
*.db_block_size=8192
*.db_create_file_dest='+DATA'
*.db_domain='ANTIBES'
*.db_file_name_convert='+DATA2/testdg_dg01','+DATA2/testdg_dg02'
*.db_name='TESTDG'
*.db_recovery_file_dest='+FRA'
*.db_recovery_file_dest_size=1782579200
*.db_unique_name='TESTDG_DG02'
*.dg_broker_start=TRUE
*.diagnostic_dest='/oracle'
*.local_listener='(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.168.56.152)(PORT=1521)))'
*.log_archive_config='dg_config=(TESTDG_DG02,testdg_dg01)'
*.log_archive_dest_1='LOCATION='+FRA VALID_FOR=(ALL_LOGFILES,ALL_ROLES)'
*.log_archive_dest_state_2='ENABLE'
*.log_archive_format='%t_%s_%r.dbf'
TESTDG.log_archive_format='%t_%s_%r.dbf'
*.log_archive_max_processes=4
*.log_archive_min_succeed_dest=1
TESTDG.log_archive_trace=0
*.open_cursors=300
*.pga_aggregate_target=367001600
*.processes=150
*.remote_login_passwordfile='EXCLUSIVE'
*.service_names='TESTDG.ANTIBES'
*.sga_target=1101004800
*.standby_file_management='AUTO'
*.undo_tablespace='UNDOTBS1'
```

Une fois le fichier créé, le convertir en spfile dans TESTDG de dataguard02 :

```
create spfile from pfile = '/tmp/initTESTDG.ora.forstdby';
```

5. Démarrage de l'instance TESTDG_DG02 en NOMOUNT

```
SQL> startup nomount
```

6. Création de la standby database avec RMAN

```
[oracle@dataguard02 trace]$ rman target sys/xxxxxxx@testdg_dg01_dg auxiliary
sys/xxxxxxx@testdg_dg02_dg
connected to target database: TESTDG (DBID=2861418430)
connected to auxiliary database: TESTDG (not mounted)

RMAN> duplicate target database for standby from active database dorecover
nofilenamecheck;
Starting Duplicate Db at 07-DEC-16
using target database control file instead of recovery catalog
allocated channel: ORA_AUX_DISK_1
channel ORA_AUX_DISK_1: SID=138 device type=DISK
contents of Memory Script:
{
backup as copy reuse
targetfile '/oracle/ora112/dbs/orapwTESTDG' auxiliary format
'/oracle/ora112/dbs/orapwTESTDG' ;
}
executing Memory Script
[...]
```

```
Starting recover at 07-DEC-16
allocated channel: ORA_AUX_DISK_1
channel ORA_AUX_DISK_1: SID=16 device type=DISK
starting media recovery
archived log for thread 1 with sequence 188 is already on disk as file
+FRA/testdg_dg02/archivelog/2016_12_07/thread_1_seq_188.299.929964369
archived log file
name=+FRA/testdg_dg02/archivelog/2016_12_07/thread_1_seq_188.299.929964369 thread=1
sequence=188
media recovery complete, elapsed time: 00:00:00
```

7. Activation de la flashback database côté physical standby database

```
alter database flashback on;
```

8. Ajout de la physical standby database dans Dataguard

```
DGMGRL> add database testdg_dg02 as connect identifier is testdg_dg02 maintained as
physical;
DGMGRL> enable database testdg_dg02
DGMGRL> show configuration
Configuration - testdg
Protection Mode: MaxPerformance
Databases:
testdg_dg01 - Primary database
testdg_dg02 - Logical standby database
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
```

9. Préparation des tables sans clé primaire pour la conversion en logical standby

Il est nécessaire de détecter toutes les tables qui ne contiennent pas de clés primaires dans la base de données primaire.

Ces tables sont considérées comme étant maintenues avec clé primaire par l'application.

Un script permet de déterminer les tables candidates à l'ajout d'une clé primaire virtuelle :

```
select owner,table_name from dba_logstdby_not_unique
where (owner,table_name) not in
(select distinct owner,table_name from dba_logstdby_unsupported)
and bad_column = 'Y'
```

Pour les tables retournées par ce script, il faudra identifier la colonne étant de clé primaire applicative et poser une clé primaire fictive sur celle-ci.

L'ajout de clé primaire fictive se fait ainsi :

```
alter table user1.table1 add primary key(colonne1) rely disable;
```

10.Préparation de l'environnement pour la conversion de la standby de physique à logique

Du côté de la physical standby database :

```
alter database recover managed standby database cancel;
```

Du côté de la primary database :

```
exec dbms_logstdby.build
```

BUILD Procedure

Ensures supplemental logging is enabled properly and builds the LogMiner dictionary

source : https://docs.oracle.com/cd/B28359_01/appdev.111/b28419/d_lsbydb.htm#i1001971

Du côté de la physical standby database :

```
alter database recover to logical standby testdg;
```

```
shutdown
startup mount
```

```
alter system set log_archive_dest_1 = 'location=use_db_recovery_file_dest
db_unique_name=testdg_dg02' scope = both;
```

11.Ouverture de la logical standby en RESETLOGS

```
alter database open resetlogs;
```

12. Démarrage de l'application des redologs en provenance de la primary database

```
alter database start logical standby apply immediate;
```

A ce point, la logical standby database est en fonctionnement nominal.

13. Enregistrement de la logical standby dans Dataguard

Dans Dataguard, la standby database est vue physical. Il faut la convertir en logical. Cette opération est réalisée en supprimant l'ancienne configuration de la base de données standby et en ajoutant la nouvelle :

```
[oracle@dataguard01 ~]$ dgmgrl
DGMGRL> connect sys/xxxxxxx
DGMGRL> show configuration
Configuration - testdg
Protection Mode: MaxPerformance
Databases:
testdg_dg01 - Primary database
testdg_dg02 - Physical standby database
Error: ORA-16810: multiple errors or warnings detected for the database
Fast-Start Failover: DISABLED
Configuration Status:
ERROR
DGMGRL> disable database testdg_dg02
Disabled.
DGMGRL> remove database testdg_dg02
Removed database "testdg_dg02" from the configuration
DGMGRL> add database testdg_dg02 as connect identifier is testdg_dg02 maintained as
logical;
Database "testdg_dg02" added
DGMGRL> enable database testdg_dg02;
Enabled.
DGMGRL> show configuration
Configuration - testdg
Protection Mode: MaxPerformance
Databases:
testdg_dg01 - Primary database
testdg_dg02 - Logical standby database
Fast-Start Failover: DISABLED
Configuration Status:
SUCCESS
```

A ce point, l'ensemble de l'architecture est en mode de fonctionnement nominal. Il devient possible de basculer (switchover) sur la logical standby database, avec les restrictions propres à ce type de standby.

Source : <https://www.champavert-consulting.fr/2016/12/07/oracle-dataguard-creer-une-logical-standby-database/>