

Table des matières

Restore en mode RAC	3
Process sessions	3
Dupliquer database	3
Listener	3
ORA-xxx	3
Divers checks	4
ORA-12514	4
ASM	4
Archivelogs	5
Recovery point	5
Pending sessions	6
Check last connections	6
Unlock users	6
Drop user connecté	6
Taille database	7
Sessions	7
FRA	8
Copie password ASM	8
Tempfile	8
Change ORACLE_HOME	9
Datapatch	9
Recompile invalid objects	9
PDBs	9
Dataguard	10
Voir ce qu'il se passe	12
Mettre le résultat d'une requête sql dans une variable	12

Restore en mode RAC

```
alter pluggable database OEM2 close immediate instances=all;

run {
set until time "to_date('04/12/2023 09:00:00', 'dd/mm/yyyy hh24:mi:ss')";
restore pluggable database OEM2;
recover pluggable database OEM2;
alter pluggable database OEM2 open resetlogs; <= don't apply archivelogs
}

alter pluggable database OEM2 open instances=all;
```

Process sessions

```
select 'alter system kill session ''' || sid || ',' || serial# || ',' || inst_id || '''; s.*
from gv$session s
where sql_id='8xdrs3uww5c0p'
and plsql_entry_object_id = 71000
and sql_exec_start < sysdate - 100/24/60/60
;
```

Dupliquer database

```
cd $ORACLE_HOME/dbs
. oraenv
echo "DB_NAME=PDBIO_R2" > initPDBIO_R2.ora
export ORACLE_SID=PDBIO_R2
sqlplus / as sysdba

startup nomount pfile='initPDBIO_R2.ora' ;

export NLS_DATE_FORMAT="DD-MON-YYYY.HH24:MI:SS"
export NLS_LANG=AMERICAN.WE8ISO8859P1
rman auxiliary / log=/home/oracle/test-restore_PDBIO.log

run
{
SET UNTIL TIME "to_date('30/10/2023 07:00:00', 'dd/mm/yyyy hh24:mi:ss')";
SET ARCHIVELOG DESTINATION TO '/u01/app/oradata/restore';
DUPLICATE DATABASE TO PDBIO_R2 BACKUP LOCATION '/BACKUP-HXL01/PDBIO/' SPFILE SET CLUSTER_DATABASE='FALSE' NOFILENAMECHECK ;
switch datafile all;
switch tempfile all;
}

select NAME,OPEN_MODE from v$database;
show pdbs ;
shutdown abort ;
startup mount exclusive restrict ;
drop database ;
```

Listener

Forcer prise en compte rapide des services BDD

```
sqlplus / as sysdba
alter system register ;
```

ORA-xxx

- ORA-01033 : check password primary ↔ standby
- ORA-16136 : checker la FRA
- ORA-16136 : ORA-46372: audit file '/u01/app/odaorabase/oracle/audit/PABIO2' : mv répertoire en .old ou rm
- Checks sémaphores : sysresv

Divers checks

```

select * from v$restore_point;

select database_role from v$databse ;

SELECT LOG_MODE FROM SYS.V$DATABASE;

set line 200
col dest_name format a45
select NAME,DATABASE_ROLE,OPEN_MODE,PROTECTION_MODE,PROTECTION_LEVEL, CURRENT_SCN,FLASHBACK_ON,FORCE_LOGGING from v$databse;

select
"Reserved_Space(GB)", "Reserved_Space(GB)" - "Free_Space(GB)" "Used_Space(GB)",
"Free_Space(GB)"
from(
select
(select sum(bytes/(1014*1024*1024)) from dba_data_files) "Reserved_Space(GB)",
(select sum(bytes/(1024*1024*1024)) from dba_free_space) "Free_Space(GB)"
from dual
);

select name
, round(space_limit / 1024 / 1024) size_mb
, round(space_used / 1024 / 1024) used_mb
, decode(nvl(space_used,0),0,0,round((space_used/space_limit) * 100)) pct_used
from v$recovery_file_dest
order by name ;

```

ORA-12514

```
SQL> show parameter service ;
```

NAME	TYPE
-----	-----
VALUE	

service_names	string
PLNAVAMP_IDC2.fr.orpea.net	

```
SQL> show parameter domain ;
```

NAME	TYPE
-----	-----
VALUE	

db_domain	string
fr.orpea.net	

```
SQL> alter system reset db_domain ;
```

System altered.

```
SQL> alter system reset db_domain scope = spfile SID='*' ;
```

```
alter system reset db_domain scope = spfile SID='*'
*
```

```
ERROR at line 1:
ORA-32010: cannot find entry to delete in SPFILE
```

ASM

- Check liste datafiles ;

```
SQL> select NAME from v$datafile ;
```

NAME

+ADATA/antarp/system-antarp-01.dbf
+ADATA/antarp/sysaux-antarp-01.dbf
+ADATA/antarp/undotbs1-antarp-01.dbf
+ADATA/antarp/users-antarp-01.dbf

```
+ADATA/antarp/antares_data-antarp-01.dbf
+ADATA/antarp/antares_indx-antarp-01.dbf
+ADATA/antarp/undotbs2-antarp-01.dbf
+ADATA/antarp/cpsure-antarp-01.dbf
+ADATA/antarp/antares_data-antarp-02.dbf
```

Archivelogs

- Voir logs

```
select to_char(timestamp,'DD-MON-YYYY HH24:MI:SS') "timestamp",MESSAGE,SEVERITY from v$dataguard_status
where SEVERITY <>'Control';
```

```
select to_char(timestamp,'DD-MON-YYYY HH24:MI:SS') "timestamp",MESSAGE,SEVERITY from v$dataguard_status ;
```

- Check réception archivelogs (à lancer sur standby) :

```
select
to_char(max(next_time), 'DD-MON-YY:HH24:MI:SS') v_Last_Received
from v$archived_log
where sequence# = (select max(sequence#) from v$archived_log);
```

- Checker config

```
set linesize 500 pages 0
col value for a120
col name for a25
select name, value
from v$parameter
where name in ('db_name','db_unique_name','log_archive_config', 'log_archive_dest_1','log_archive_dest_2','log_archive_dest_3','log_archive_dest_4',
'log_archive_dest_state_1','log_archive_dest_state_2','log_archive_dest_state_3', 'log_archive_dest_state_4','remote_login_passwordfile',
'log_archive_format','log_archive_max_processes','fal_server','fal_client',
'standby_file_management');
/
```

```
SQL> ARCHIVE LOG LIST ;
Database log mode          Archive Mode
Automatic archival        Enabled
Archive destination       USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 41817
Next log sequence to archive 41819
Current log sequence       41819
```

```
SQL> SHOW PARAMETER DB_RECOVERY_FILE_DEST ;
db_recovery_file_dest      string          +FPFRA
db_recovery_file_dest_size big integer     152G
```

```
SQL> select dest_name,status,destination from V$ARCHIVE_DEST;
LOG_ARCHIVE_DEST_1                                VALID
USE_DB_RECOVERY_FILE_DEST
```

- Modif config

```
alter system set fal_server='frantarp_1dc2' scope=both;
ALTER SYSTEM SET LOG_ARCHIVE_DEST_STATE_3 = DEFER SCOPE=BOTH;
alter system reset log_archive_dest_2 sid='*';
alter system set log_archive_config='DG_CONFIG=(FRANTARP_1DC1,FRANTARP_1DC2)' scope=both;
alter system set LOG_ARCHIVE_DEST_2='SERVICE=frantarp_1dc2 SYNC NOAFFIRM VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=frantarp_1dc2' scope=both sid='*';
ALTER SYSTEM SET log_archive_dest_3 = '';
```

Recovery point

```
ALTER DATABASE FLASHBACK ON;
create restore point BEFORE_MIG GUARANTEE FLASHBACK DATABASE;
```

```
select * from v$restore_point;
```

```
RMAN> LIST RESTORE POINT ALL;
```

```
using target database control file instead of recovery catalog
```

```
SCN          RSP Time   Type      Time      Name
-----
```



```

2 PDB$SEED          READ ONLY NO
3 TFRHN001          READ WRITE NO
4 TFRHN001Q         READ WRITE NO
5 TFRMPY01          READ WRITE NO
SYS@TBBI01> ALTER SESSION set container=TFRMPY01;

SYS@TBBI01> show pdbs ;

  CON_ID CON_NAME          OPEN MODE RESTRICTED
-----
5 TFRMPY01          MOUNTED
SYS@TBBI01> alter pluggable database TFRMPY01 open restricted;

Pluggable database altered.

SYS@TBBI01> show pdbs ;

  CON_ID CON_NAME          OPEN MODE RESTRICTED
-----
5 TFRMPY01          READ WRITE YES
SYS@TBBI01> ALTER SESSION set container=TFRMPY01;

Session altered.

SYS@TBBI01> DROP USER FRMPY01 CASCADE ;

```

Taille database

```

select
"Reserved_Space(GB)", "Reserved_Space(GB)" - "Free_Space(GB)" "Used_Space(GB)",
"Free_Space(GB)"
from(
select
(select sum(bytes/(1014*1024*1024)) from dba_data_files) "Reserved_Space(GB)",
(select sum(bytes/(1024*1024*1024)) from dba_free_space) "Free_Space(GB)"
from dual
);

```

```
select sum (bytes)/1024/1024/1024 size_GB from dba_segments ;
```

```

WITH
TS_ALLOC as
(
select TABLESPACE_NAME TS, round(sum(bytes/1024/1024/1024)) ALLOC
from cdb_data_files
group by TABLESPACE_NAME
union
select TABLESPACE_NAME TS, round(sum(bytes/1024/1024/1024)) ALLOC
from cdb_temp_files
group by TABLESPACE_NAME
),
TS_USED as
(
select TABLESPACE_NAME TS, round(sum(bytes/1024/1024/1024)) FREE
from cdb_free_space
group by TABLESPACE_NAME
)
select sum(ALLOC) "ALLOC (G)", sum(FREE) "FREE (G)", sum(ALLOC) - sum(FREE) "USED (G)"
from TS_ALLOC a LEFT OUTER JOIN TS_USED u on a.TS=u.TS ;

```

Sessions

```

set echo on time on timing on lines 200 pages 1000
col username format a30
col machine format a30
col service_name format a30
col logon_time format a20
select
username
--
, machine
, inst_id
, service_name

```

```

, status
, to_char(logon_time,'YYYY/MM/DD HH24:MI') logon_time
, count(1) sessions_cnt
from
gv$session
where
l=1
and type='USER'
and service_name not like 'SYS%'
and username not like 'SYS%'
and username not like 'PUBLIC'
group by
username
--
, machine
, inst_id
, service_name
, status
, to_char(logon_time,'YYYY/MM/DD HH24:MI')
order by
sessions_cnt desc;

```

```

select sid,serial# from v$session where username = 'FRMPY01' ;
select 'alter system kill session ''' || sid || ',' || serial# || ''' immediate;' from gv$session where username ='FRMPY01';

```

- RAC

```

select sid,serial#,inst_id from gv$session where username = 'FRMPY01' ;
select 'alter system kill session '''||sid||','||serial#||','||inst_id||''' immediate;' as script from gv$session where username='FRMPY01' ;

```

FRA

```

SYS@TEBI01> show parameter recovery

```

NAME	TYPE
-----	-----
VALUE	
-----	-----
db_recovery_file_dest	string
+RECO	
db_recovery_file_dest_size	big integer
200G	
recovery_parallelism	integer
0	
remote_recovery_file_dest	string

```

ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE=400g SCOPE=BOTH ;

```

```

select name
, round(space_limit / 1024 / 1024) size_mb
, round(space_used / 1024 / 1024) used_mb
, decode(nvl(space_used,0),0,0,round((space_used/space_limit) * 100)) pct_used
from v$recovery_file_dest
order by name ;

```

Copie password ASM

```

orapwd file=orapwt2lcbio password=Dedalus1 ignorecase=n format=12 force=Y
asmcmd pwcopy --dbuniquename T21CBI0 /u01/app/odaorahome/oracle/product/19.0.0.0/dbhome_5/dbs/orapwt2lcbio +DATA/T21CBI0/orapwt2lcbio

```

```

srvctl modify database -d ORCL -pwfile +DATA/DRORCL/PASSWORD/pwdorcl

```

Tempfile

```

SYS@PDBI01> select name from v$tempfile;

```

NAME

+DATA/P21DBI0/TEMPFILE/temp.1823.1125772463
+DATA/P21DBI0/CB949F6CD1D60A81E0533E60180A9ACE/TEMPFILE/temp.1824.1125772463
+DATA/P21DBI0/CB94C1145D745969E0533E60180A9064/TEMPFILE/temp.1826.1125772465

```
+DATA/P21DBI0/CCE2ECA49E5F05B5E0533E60180A5B40/TEMPFILE/temp.1825.1125772465
+DATA/P21DBI0/D58B172F55E0EB56E0533E60180A7510/TEMPFILE/temp.1852.1131484295
```

Change ORACLE_HOME

```
oda1db0host# srvctl stop database -db P21CPAT
oda1db0host# srvctl modify database -d P21CPAT -oraclehome "/u01/app/odaorahome/oracle/product/19.0.0.0/dbhome_5"
oda1db0host# srvctl start database -db P21CPAT
```

Datapatch

```
oda2db0host# . oraenv
oda2db0host# cd $ORACLE_HOME && ./OPatch/datapatch -db PHBI01 -verbose
```

Recompile invalid objects

```
@$ORACLE_HOME/rdbms/admin/utlrp.sql
```

PDBs

- show

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	YES
3	PFRCHA02	READ WRITE	NO
4	PFRCHA02Q	READ WRITE	NO
5	PFRNPC01	READ WRITE	NO

```
SQL>
```

- change mode

```
SQL> alter pluggable database pdb$seed open read only ;
```

```
Warning: PDB altered with errors.
```

- cd to PDB

```
ALTER SESSION SET CONTAINER=PFRNPC01;
```

- Restricted

Check si datapatch OK

```
SQL> show pdbs
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	PHRQ_OEM	READ WRITE	NO
4	PHQR_OEM1	READ WRITE	YES
5	PHQR_OAS1	READ WRITE	YES

```
SQL> alter pluggable database PHQR_OEM1 open force ;
```

```
Pluggable database altered.
```

```
SQL> show pdbs ,
```

CON_ID	CON_NAME	OPEN MODE	RESTRICTED
2	PDB\$SEED	READ ONLY	NO
3	PHRQ_OEM	READ WRITE	NO
4	PHQR_OEM1	READ WRITE	NO
5	PHQR_OAS1	READ WRITE	YES

```
SQL> alter pluggable database PHQR_OAS1 open force ;
```

```
SQL> alter session set container=PHQR_OEM1 ;
Session altered.
SQL> alter system disable restricted session;
```

Dataguard

- start Redo Apply

```
ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
```

- Switch manuel

```
## Check if possible and Convert Primary to Physical Standby
```

```
srvctl stop instance -d FRTLPM_PRM -i TLM2
sqlplus / as sysdba
SELECT SWITCHOVER_STATUS FROM V$DATABASE; (must be TO_STANDBY)
ALTER DATABASE COMMIT TO SWITCHOVER TO STANDBY WITH SESSION SHUTDOWN;
```

```
## Check if possible and Convert the Standby to Primary
```

```
sqlplus / as sysdba
SELECT SWITCHOVER_STATUS FROM V$DATABASE; (must be TO_PRIMARY)
ALTER DATABASE COMMIT TO SWITCHOVER TO PRIMARY WITH SESSION SHUTDOWN;
```

```
## Deactivate all STANDBY Database and Open Primary
```

```
alter system set log_archive_dest_state_2=defer scope=both ;
alter system set log_archive_dest_state_3=defer scope=both ;
alter system set log_archive_dest_state_4=defer scope=both ;
alter database open ;
```

```
# Restart old PRIMARY and Start Standby Mode
```

```
startup mount
ALTER DATABASE RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE DISCONNECT;
```

```
# Modify CLUSTER Configuration
```

```
srvctl modify database -d FRTLMP_IDC1 -r PRIMARY -s OPEN
```

- Switch semi-manuel

```
kill du switchover ;
stop de la base ;
srvctl stop database -db FRINTEGP_PRM
restart des bases de l'autre côté
```

```
srvctl modify database -db frintegp_idc1 -role PRIMARY -startoption OPEN
stop /start
```

```
startup mount de l'ancienne primary
check tnsping
remove configuration ;
```

```
CREATE CONFIGURATION drintegp AS PRIMARY DATABASE IS frintegp_idc1 CONNECT IDENTIFIER IS frintegp_idc1;
ADD DATABASE frintegp_idc2 AS CONNECT IDENTIFIER IS frintegp_idc2;
ENABLE CONFIGURATION;
```

- Logs du broker : drc* file
- Check de la conf du broker

```
SQL> show parameter broker ;
```

NAME	TYPE
connection_brokers	string
((TYPE=DEDICATED)(BROKERS=1)),	
((TYPE=EMON)(BROKERS=1))	
dg_broker_config_file1	string

```
+FPDAT/FRSIRHAP_IDC2/dg_config
-sirhap-01.dbf
dg_broker_config_file2      string
+FPDAT/FRSIRHAP_IDC2/dg_config
-sirhap-02.dbf
```

```
NAME                          TYPE
-----
VALUE
-----
dg_broker_start                boolean
TRUE
use_dedicated_broker          boolean
FALSE
```

- Restart du broker

```
SQL> alter system set dg_broker_start=FALSE scope=BOTH SID='*' ;
SQL> alter system set dg_broker_start=TRUE scope=BOTH SID='*' ;
```

- Switchover

```
dgmgrl sys@P72HBIO
show configuration
validate database <STANDBY>
show database <STANDBY>
show database verbose <STANDBY>
```

- Recréer config

```
remove configuration ;
CREATE CONFIGURATION my_dg_config AS PRIMARY DATABASE IS db1lg CONNECT IDENTIFIER IS db1lg;
ADD DATABASE db1lg_stby AS CONNECT IDENTIFIER IS db1lg_stby;
ENABLE CONFIGURATION;
```

- Check lag

```
col NAME format a10
select NAME,TIME,UNIT,COUNT,LAST_TIME_UPDATED from V$STANDBY_EVENT_HISTOGRAM
where name like '%lag' and count >0 order by LAST_TIME_UPDATED;
```

```
select name, value, time_computed, datum_time from v$dataguard_stats where name='%lag';
```

Primary

```
SQL> select scn_to_timestamp(current_scn) from v$database;
```

```
select sysdate,database_mode,recovery_mode, gap_status
from v$archive_dest_status
where type='PHYSICAL'
and gap_status !='NO GAP';
```

```
select *
from v$dataguard_status
where severity in ('Error','Fatal')
and timestamp > (sysdate -1);
```

```
select sysdate,status,error
from gv$archive_dest_status
where type='PHYSICAL'
and status!='VALID'
or error is not null;
```

Standby

```
col name format a45
select name,value,time_computed,datum_time
from v$dataguard_stats
where name='transport lag'
and value > '+00 00:01:00';
```

```
col name format a45
select name,value,time_computed,datum_time
from v$dataguard_stats
where name='apply lag'
```

```
and value > '+00 00:01:00';

select max(timestamp)
  from gv$recovery_progress group by inst_id;

set line 500 pages 9999
col severity form a40
col message form a131
select SEVERITY,to_char(timestamp,'DD-MON-YYYY HH24:MI:SS') "timestamp",MESSAGE from v$dataguard_status;
select distinct error from v$archive_dest;
```

- Check réception archives logs

```
select to_char(timestamp,'DD-MON-YYYY HH24:MI:SS') "timestamp",MESSAGE,SEVERITY from v$dataguard_status
where SEVERITY <>'Control';
```

```
select
  to_char(max(next_time), 'DD-MON-YY:HH24:MI:SS') v_Last_Received
  from v$archived_log
 where sequence# = (select max(sequence#) from v$archived_log);
```

<https://docs.oracle.com/en/database/oracle/oracle-database/19/haovw/monitor-oracle-data-guard-configuration.html#GUID-51E71BB5-EE63-434A-976B-AE89C807A405>
http://www.datadisk.co.uk/html_docs/oracle_dg/cheatsheet.htm

Voir ce qu'il se passe

```
set head off pages 0 lines 120
select p.spid,s.sid, s.serial#, substr(s.username,1,10)||', '||process,
s.program,s.module,s.status, osuser ,
buffer_gets, disk_reads, executions,users_executing, first_load_time,'*',s.wait_class
,a.rows_processed, a.sql_id, sql_text -- sql_fulltext
from v$process p, v$session s, v$sqlarea a
where a.address=s.sql_address
and p.addr=s.paddr
and users_executing > 0
and s.status='ACTIVE'
/
```

Mettre le résultat d'une requête sql dans une variable

```
vMaxSeqApply=`sqlplus -s "/" as sysdba" <<EOF
set head off
set PAGES 0
set FEED off
select substr(max(sequence#),0) from v\$\archived_log where applied='YES' and DEST_ID=2 group by THREAD#;
exit
EOF
`
```

From:
<https://unix-bck.ndlp.info/> - **Where there is a shell, there is a way**

Permanent link:
https://unix-bck.ndlp.info/doku.php/informatique:base_de_donnees:tips?rev=1701703286

Last update: **2023/12/04 16:21**