

# **Flashback Database ile Standby Üzerinden Veri Kurtarmak**

## İçerik

1. Physical Standby üzerinde Flashback Database.....	3
2. Test Ortam(lar)ını Oluşturmak.....	4
3. Physical Standby Veritabanını Oluşturmak.....	5
4. Physical Standby Üzerinde Flashback Özelliğini Açmak.....	9
5. Flashback Standby Database ile Veri Kurtarma.....	10
6. Standby Veritabanını Eski Hâline Alma.....	13
7. Kaynaklar.....	15

## 1. Physical Standby üzerinde Flashback Database

Yaklaşık 7 yıl evvel Flashback Database özelliğinden bahsetmiştim. (Yazdığım bu yazıya [Oracle Flashback Database Teknolojisi](#) bağlantısından erişebilirsiniz.) Aradan geçen yedi yılda birçok defa kullandım ve bolca tavsiye ettim.

Flashback Database, özellikle test ortamları için harika bir çözümdü. Örneğin bankacılık paketi için günsonu uygulamasının baştan yazıldığını düşünün. Tekrar tekrar 'temiz' veriyle test yapmanız gerekecektir. Her günsonu testi, elinizdeki veriyi bozacaktır. Bu gibi durumlar için Flashback Database kullanımı süreçleri çok kolaylaştırır. Yoğun bir günsonu süreci çalıştırıp, test sonuçlarını aldıktan sonra, veritabanını eski hâline çok kısa bir sürede dönebilirsiniz. Bu teknoloji olmasaydı, her seferinde yedekten dönmek hem efor hem de zaman bakımından çok maliyetli olurdu.

Flashback Database özelliğini, üretim (prod) ortamlarında kullanmak da mümkün. Ama bir problem durumunda üretim ortamını kapatıp, birkaç saat geriye çekmek bana hiçbir zaman mantıklı gelmemişti. Bu nedenle bir alternatif düşünüp, cin bir fikirle ortaya çıkmıştım: *Flashback Database'i canlı sistemlerde kullanmak yerine, standby sistemlerimde kullanabilirdim.* Bir problem durumunda da standby ortama replikasyonu keser; flashback ile standby ortamı geçmişe çeker ve istediğim veriyi bir şekilde dışarı alabilirdim. Sonrasında bu veriyi canlı sisteme aktarmak çocuk oyuncağıydı. Fikir güzel olsa da, 10g üzerinde testlerim pek başarılı sonuç vermedi. Bir süre sonra çalışıyor fakat sonra sorun çıkıyordu; istediğim zamana dönemiyordum. <sup>1</sup>

Aradan birkaç yıl geçti ve 11g sürümüyle birlikte, standby üzerinde flashback kullanım problemleri büyük oranda düzeldi. Bu sürümden itibaren standby ortamlarda, flashback kullanımını özellikle tavsiye etmeye başladım. Bu yazıda bu konuya değineceğim. Fakat benzer bir konu olan "Snapshot Standby Database" bu yazı içinde olmayacak. Onu konuya da belki bir başka yazıda değiniriz.

---

<sup>1</sup> Benim 10g sürümlü Standby üzerindeki Flashback kullanımım pek başarılı olmasa da, güzel sonuçlar alanlar da var. Konuyla ilgili [How to Implement a "snapshot standby" Using Oracle Data Guard 10 g Release 2](#) makalesini inceleyebilirsiniz.

## 2. Test Ortam(lar)ını Oluşturmak

Bu testi yapabilmek için iki adet makineye ihtiyacımız var. Bu makinelerden birincisi üretim (prod) olarak hizmet ederken, diğeri standby rolünü üstlenecek. Testlerim esnasında VirtualBox üzerinde oluşturduğum sanal makineleri kullandım. Öncelikle bir sanal makine oluşturup, Oracle Linux 6.5 işletim sistemi ve 12c veritabanı kurdum. İkinci makineyi baştan kurmakla vakit kaybetmedim. Bunun yerine, ilk oluşturduğum makineden kopyasını (clone) aldım. Burada işletim sistemi, veritabanı kurulumu vb. konulara değinmeyeceğim. Fakat oluşturduğum clone ile ilgili bir iki ufak önerim olacak. İlki clone alırken, MAC adreslerini değiştirmeniz yönünde... Ardından, makineyi açıp ismini değiştirmenizi ve buna göre düzenlemeleri yapmanızı tavsiye ederim. Ben kabaca bu adımları paylaşayım:

```
# ESKI MAKINE ADI KONTROL EDILIR VE YENI ISIM OLARAK 'stbyvm' VERILIR
[root@testvm ~]# hostname
testvm
[root@stbyvm ~]# hostname stbyvm

# /etc/sysconfig/network DOSYASINDA, MAKINE ADI DEGISTIRILIR
[root@stbyvm ~]# cat /etc/sysconfig/network
NETWORKING=yes
HOSTNAME=stbyvm

# hosts DOSYASINDA, MAKINE ADI ICIN DUZENLENIR
[root@testvm ~]# cat /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
127.0.0.1 stbyvm
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6

# REBOOT ETMEDEN DEGISEN MAKINE ADINI GECERLI HALE GETIRMEK MUMKUNDUR
[root@testvm ~]# service network restart

# ETHERNET KART ISIMLERI DUZENLENIR
[root@testvm ~]# cat /etc/udev/rules.d/70-persistent-net.rules
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:3a:62:18", ATTR{type}=="1",
KERNEL=="eth*", NAME="eth0"

# PCI device 0x8086:0x100e (e1000)
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="08:00:27:16:e2:5b", ATTR{type}=="1",
KERNEL=="eth*", NAME="eth1"
```

Bu işlemlere ek olarak, ben ethernet kartlarına statik IP'ler atamıştım. Siz de /etc/sysconfig/network-scripts altındaki konfigürasyon script'leriyle statik IP'leri ayarlayabilirsiniz. Bütün düzenlemeleri tamamlayıp, clone sisteminizi yeniden başlatın.

### 3. Physical Standby Veritabanını Oluşturmak

Öncelikle üretim ortamını, arşiv moda çekmemiz, ardından bir yedek almamız son olarak da bu yedeği standby olacak biçimde dönmemiz gerekiyor. Veritabanı dosyaları üretim ve yedek ortamlarda farklı klasör ve isimlere sahip olabilir. Yalnız kolaylık olması açısından, ben aynı klasör ve dosya isimlerini kullanacağım.

```
-- PRIMARY VERITABANINI ARCHIVE MODE'A CEKME ASAMALARI
CDBA:SQL> startup mount
CDBA:SQL> alter system set log_archive_dest_1='LOCATION=/media/SharedFS/BACKUP/archives';
CDBA:SQL> alter system set log_archive_format='cdba_%t_%s_%r.arc' scope=spfile;
CDBA:SQL> alter system set archive_lag_target=1800;
CDBA:SQL> alter database archive log;
-- ASAGIDAKI ADIM ZORUNLU DEGIL AMA DATA GUARD ICIN YARARLI OLACAKTIR
CDBA:SQL> alter database force logging;
CDBA:SQL> alter database open;
```

```
-- tnsnames.ora DOSYASI DUZENLENIYOR
[oracle@testvm admin]$ cat tnsnames.ora
-- # Generated by Oracle configuration tools.
--
-- LISTENER_CDBA =
--   (DESCRIPTION_LIST =
--     (DESCRIPTION =
--       (ADDRESS = (PROTOCOL = TCP)(HOST = testvm)(PORT = 1526))
--       (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1526))
--     )
--   )
--
-- CDBA_PROD =
--   (DESCRIPTION =
--     (ADDRESS = (PROTOCOL = TCP)(HOST = testvm)(PORT = 1526))
--     (CONNECT_DATA =
--       (SERVER = DEDICATED)
--       (SERVICE_NAME = CDBA)
--     )
--   )
--
-- CDBA_STBY =
```

```

-- (DESCRIPTION =
--   (ADDRESS = (PROTOCOL = TCP)(HOST = stbyvm)(PORT = 1526))
--   (CONNECT_DATA =
--     (SERVER = DEDICATED)
--     (SERVICE_NAME = CDBA)
--   )
-- )

-- TNS DOSYASI KOPYALANDIKTAN SONRA,
-- listener.ora VE tnsnames.ora DOSYALARINDA STANDBY ICIN DUZENLEME YAPILIR
[oracle@testvm admin]$ scp tnsnames.ora \
    stbyvm:/u03/app/oracle/product/12.1.0.2/dbhome_1/network/admin

-- oracle@stbyvm's password:
-- tnsnames.ora                               100% 925      0.9KB/s   00:00

-- PAROLA DOSYASI KOPYALANIR
[oracle@testvm dbs]$ scp orapwCDBA stbyvm:/u03/app/oracle/product/12.1.0.2/dbhome_1/dbs
-- oracle@stbyvm's password:
-- orapwCDBA                                  100% 7680     7.5KB/s   00:00

-- PFILE PROD ORTAMDAN ALINIR VE STANDBY'A GORE DUZENLENIR
PROD:SQL> create pfile='/tmp/create_pfile.text' from spfile;
[oracle@testvm ~]$ cat /tmp/create_pfile.text
*.archive_lag_target=1800
*.audit_file_dest='/u03/app/oracle/admin/CDBA/adump'
*.audit_trail='db'
*.compatible='12.1.0.2.0'
*.control_files='/u03/app/oracle/oradata/CDBA/control01.ctl','/u03/app/oracle/oradata/CDBA/control02.ctl'
*.db_block_size=8192
*.db_domain=''
*.db_name='CDBA'
*.diagnostic_dest='/u03/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=CDBAXDB)'
*.enable_pluggable_database=true
*.inmemory_size=268435456
*.local_listener='LISTENER_CDBA'
*.log_archive_dest_1='LOCATION=/media/SharedFS/BACKUP/archives'
*.log_archive_format='cdba_%t_%s_%r.arc'
*.open_cursors=300
*.pga_aggregate_target=390m

```

```
*.processes=300
```

```
*.remote_login_passwordfile='EXCLUSIVE'
```

```
*.sga_max_size=2147483648
```

```
*.sga_target=2147483648
```

```
*.undo_tablespace='UNDOTBS1'
```

```
[oracle@testvm ~]$ scp /tmp/create_pfile.text stbyvm:/tmp
```

```
-- oracle@stbyvm's password:
```

```
-- create_pfile.text                               100% 1110      1.1KB/s   00:00
```

```
-- PFILE'DAN SPFILE OLUSTURULUR
```

```
STBY:SQL> create spfile from pfile='/tmp/create_pfile.text';
```

```
# VERİTABANINI SIKİSTİRİP YEDEKLER
```

```
PROD:RMAN> BACKUP FULL AS COMPRESSED BACKUPSET DATABASE SKIP INACCESSIBLE INCLUDE CURRENT  
CONTROLFILE FILESPERSET 20 FORMAT '/media/SharedFS/BACKUP/rman_bck/%d_BACKUP_%T_%s.rman';
```

```
-- ...
```

```
-- piece handle=/media/SharedFS/BACKUP/rman_bck/CDBA_BACKUP_20150523_74.rman tag=TAG20150523T102848  
comment=NONE
```

```
-- channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
```

```
-- Finished backup at 23-MAY-15
```

```
RMAN> BACKUP CURRENT CONTROLFILE FOR STANDBY
```

```
  FORMAT '/media/SharedFS/BACKUP/rman_bck/%d_STBYCNTRL_%T_%s.rman';
```

```
-- ...
```

```
-- Starting Control File and SPFILE Autobackup at 23-MAY-15
```

```
-- piece handle=/u03/app/oracle/product/12.1.0.2/dbhome_1/dbs/c-4156799750-20150523-02 comment=NONE
```

```
-- Finished Control File and SPFILE Autobackup at 23-MAY-15
```

```
-- RMAN İLE STANDBY OLUSTURULUR
```

```
STBY:SQL> startup nomount;
```

```
[oracle@stbyvm ~]$ rman target sys/welcome1@CDBA_PROD auxiliary / msglog duplct.log <<EOF
```

```
run {
```

```
  allocate channel c1 type disk;
```

```
  allocate auxiliary channel cr1 type disk;
```

```
  duplicate target database for standby backup location '/media/SharedFS/BACKUP/rman_bck/'  
  nofilenamecheck dorecover;
```

```
  release channel c1;
```

```
  release channel cr1;
```

```
}
```

```
EOF
```

```
-- STANDBY TARAFINDA STANDBY REDOLOG'LAR EKLENECEK
-- (maximum number of logfiles for each thread + 1) * maximum number of threads
STBY:SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u02/app/oradata/CDBA/stbyredo01.log') SIZE 50M;
STBY:SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u02/app/oradata/CDBA/stbyredo02.log') SIZE 50M;
STBY:SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u02/app/oradata/CDBA/stbyredo03.log') SIZE 50M;
STBY:SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u02/app/oradata/CDBA/stbyredo04.log') SIZE 50M;
```

```
-- ARCHIVE LOG DOSYALARI PROD ILE FARKLI LOKASYONLARDA OLSUN DIYE YAPILDI
STBY:SQL> alter system set log_archive_dest_1='LOCATION=/media/SharedFS/BACKUP/stby_archives';
STBY:SQL> ALTER SYSTEM SET STANDBY_FILE_MANAGEMENT=AUTO;
STBY:SQL> RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE DISCONNECT FROM SESSION;
```

```
-- DATA GUARD'IN KULLANMASI ICIN ARCHIVE LOG DESTINATION'I PROD UZERINDE AYARLANIYOR
PROD:SQL> ALTER SYSTEM SET log_archive_dest_10='SERVICE=CDBA_STBY ASYNC REOPEN=60';
```

```
-- OPSİYONEL OLARAK ACTIVE DATA GUARD YAPILANDIRMASI ICIN ASAGIDAKI ISLEMLER YAPILIR2
STBY:SQL> RECOVER MANAGED STANDBY DATABASE CANCEL;
STBY:SQL> ALTER DATABASE OPEN READ ONLY;
STBY:SQL> RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE DISCONNECT FROM SESSION;
STBY:SQL> ALTER PLUGGABLE DATABASE MYPDB OPEN READ ONLY;
STBY:SQL> ALTER SESSION SET CONTAINER=MYPDB;
```

```
-- MYPDB VERİTABANI SORGULANABİLİR HALE GELİYOR
STBY:SQL> select count(*) from system.deneme444;
-- COUNT(*)
-- -----
--          90776
```

```
-- PROD VE STANDBY VERİTABANLARININ DURUMLARI KONTROL EDİLİYOR
PROD:SQL> select protection_mode, protection_level, database_role, switchover_status from
v$database;
```

```
-- PROTECTION_MODE      PROTECTION_LEVEL      DATABASE_ROLE      SWITCHOVER_STATUS
-- -----
-- MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PRIMARY            NOT ALLOWED
```

```
STBY:SQL> select protection_mode, protection_level, database_role, switchover_status from
v$database;
```

```
-- PROTECTION_MODE      PROTECTION_LEVEL      DATABASE_ROLE      SWITCHOVER_STATUS
-- -----
-- MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PHYSICAL STANDBY  NOT ALLOWED
```

2 Flashback Standby Database, Enterprise Edition veritabanlarında ek bir ücret olmadan kullanılabilir. Ancak testler esnasında kullanılan Multi-Tenant ve Active Data Guard opsiyonları ek lisans gerektirir.



```
-- BIRKAC ARCHIVELOG DOSYASI OLUSTURULUP, YANSIMASI KONTROL EDILIR
-- BU AMACLA ASAGIDAKI SORGU HER IKI ORTAMDA DA CALISTIRILIR
SQL> SELECT MAX(SEQUENCE#) FROM V$LOG_HISTORY;
```

## 4. Physical Standby Üzerinde Flashback Özelliğini Açmak

Önceki adımda, üretim ortamının yedeğini alıp, bu yedekle standby veritabanını ayağa kaldırdık. Şimdi de standby veritabanı üzerinde flashback database özelliğini aktive edeceğiz.

```
-- FLASHBACK DATABASE AKTIVASYONU ICIN UC (3) PARAMETRE DEGISIKLIGI STANDBY UZERINDE YAPILIR
-- TESTIMIZ ICIN 10 SAAT (600 DAKIKA) VE 20GB DEPOLAMA ALANIYLA FLASHBACK AYARLANIR
-- FLASHBACK ICIN KULLANILACAK LOKASYON ISE 'DB_RECOVERY_FILE_DEST' PARAMETRESINDE BELIRTILIR
STBY:SQL> ALTER SYSTEM SET DB_FLASHBACK_RETENTION_TARGET=600;
STBY:SQL> ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE=20G;
STBY:SQL> ALTER SYSTEM SET DB_RECOVERY_FILE_DEST='/media/SharedFS/BACKUP/flashback_area';
```

```
-- SIMDI DE FLASHBACK OZELLIGINI AKTIF HALE GETIRIYORUZ
STBY:SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;
STBY:SQL> RECOVER MANAGED STANDBY DATABASE CANCEL;
STBY:SQL> ALTER DATABASE FLASHBACK ON;
```

```
-- TEST AMACIYLA STANDBY VERITABANINI KAPATIP, ACIYORUZ
STBY:SQL> SHUTDOWN IMMEDIATE
STBY:SQL> STARTUP NOMOUNT
STBY:SQL> ALTER DATABASE MOUNT STANDBY DATABASE;
STBY:SQL> ALTER DATABASE OPEN READ ONLY;
STBY:SQL> ALTER PLUGGABLE DATABASE MYPDB OPEN READ ONLY;
STBY:SQL> RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE DISCONNECT FROM SESSION;
```

```
/* STANDBY UZERINDEKI FLASHBACK KONTROLU */
```

```
STBY:SQL> SELECT NAME, OPEN_MODE, FLASHBACK_ON, LOG_MODE FROM V$DATABASE;
```

```
-- NAME          OPEN_MODE          FLASHBACK_ON          LOG_MODE
-- -----
-- CDBA          READ ONLY WITH APPLY YES          ARCHIVELOG
```

```

STBY:SQL> select name AS PARAMETER_NAME, value from v$parameter
where upper(name) in ('DB_FLASHBACK_RETENTION_TARGET','DB_RECOVERY_FILE_DEST_SIZE',
                    'DB_RECOVERY_FILE_DEST','LOG_ARCHIVE_DEST_1');

-- PARAMETER_NAME          VALUE
-- -----
-- log_archive_dest_1      LOCATION=/media/SharedFS/BACKUP/stby_archives
-- db_recovery_file_dest    /media/SharedFS/BACKUP/flashback_area
-- db_recovery_file_dest_size 21474836480
-- db_flashback_retention_target 600

/* FLASHBACK DOLULUK ORANINI GOSTERIR */
SET LINES 130 FEEDBACK OFF TRIMSPOOL ON
COLUMN NAME FORMAT A40
select name,
       to_char(space_limit, '999,999,999,999') as space_limit,
       to_char(space_limit - space_used + space_reclaimable, '999,999,999,999') as space_available,
       round((space_used - space_reclaimable)/space_limit * 100, 1) as pct_full
from   v$recovery_file_dest;

-- NAME                                SPACE_LIMIT      SPACE_AVAILABLE    PCT_FULL
-- -----
-- /media/SharedFS/BACKUP/flashback_area 21,474,836,480   21,404,008,448     .3

/* FLASHBACK ILE GIDILEBİLECEK EN ESKI ZAMAN */
SELECT
       TO_CHAR( OLDEST_FLASHBACK_SCN ) SCN,
       TO_CHAR(OLDEST_FLASHBACK_TIME, 'HH24:MI:SS DD/MM/YYYY') OLDEST_TIME
FROM   V$FLASHBACK_DATABASE_LOG;

-- SCN                                OLDEST_TIME
-- -----
-- 3101393                             11:31:19 23/05/2015

```

## 5. Flashback Standby Database ile Veri Kurtarma

Bütün hazırlıklarımızı yaptık. Artık bir felaket senaryosunu test edebiliriz. Kritik bir tablonun üretim ortamında yanlışlıkla "*DROP TABLE... PURGE*" komutuyla silindiğini düşünelim. Bu hatalı işlem 11:42'de gerçekleşmiş olsun. Bu bilgiler ışığında Standby üzerindeki flashback özelliğiyle bu tabloyu kurtmaya çalışalım.

```

-- DROP ISLEMINDEN ONCE HEM PROD HEM DE STANDBY VERITABANINDA
-- SYSTEM.DENEME444 TABLOSUNA ERISEBILİYORUZ
STBY:SQL> select count(*) from system.deneme444;
-- COUNT(*)
-- -----
--      90939

-- PROD ORTAMINDA TABLO, 11.42'DE DROP EDILIR
PROD:SQL> drop table system.deneme444 purge;

-- YAPILAN ISLEM STANDBY ORTAMA YANSIR. BU NEDENLE SORGU ARTIK HATA ALIR
STBY:SQL> select count(*) from system.deneme444;
-- select count(*) from system.deneme444
--
--          *
-- ERROR at line 1:
-- ORA-00942: table or view does not exist

-- STANDBY VERITABANINI TEKRAR STANDBY YAPABİLMEK İCİN BİR 'RESTORE POINT' OLUSTURUYORUZ
STBY:SQL> RECOVER MANAGED STANDBY DATABASE CANCEL;
STBY:SQL> CREATE RESTORE POINT before_standby_activation GUARANTEE FLASHBACK DATABASE;

STBY:SQL> SELECT NAME, SCN, STORAGE_SIZE FROM V$RESTORE_POINT
WHERE GUARANTEE_FLASHBACK_DATABASE = 'YES';
-- NAME                                SCN STORAGE_SIZE
-- -----
-- BEFORE_STANDBY_ACTIVATION            3104772      52428800

STBY:SQL> FLASHBACK STANDBY DATABASE TO TIMESTAMP to_date('2015-05-23 11:41:19','YYYY-MM-DD
HH24:MI:SS');
-- Flashback complete.
STBY:SQL> ALTER DATABASE ACTIVATE STANDBY DATABASE;

-- BUTUN SURECI ALERTLOG UZERINDEN DE TAKIP EDEBİLİRİZ
[oracle@stbyvm ~]$ tail -f /u03/app/oracle/diag/rdbms/cdba/CDBA/trace/alert_CDBA.log
-- ...
-- Incomplete Recovery applied until change 3102631 time 05/23/2015 11:41:20
-- Flashback Media Recovery Complete
-- Completed: FLASHBACK STANDBY DATABASE TO TIMESTAMP to_date('2015-05-23 11:41:19','YYYY-MM-DD

```

```

HH24:MI:SS')
-- Sat May 23 12:03:32 2015
-- RFS[1]: Possible network disconnect with primary database
-- ...
-- AUDIT_TRAIL initialization parameter is changed back to its original value as specified in the
parameter file.
-- ACTIVATE STANDBY: Complete - Database mounted as primary
-- Completed: alter database activate standby database

-- FLASHBACK İLE STANDBY VERİTABAN İSTENEN ZAMANA DONMUSTUR
-- BU SAYEDE ÜRETİM ORTAMINDA TABLO VERİSİNE ERİSİLEBİLİR
STBY:SQL> ALTER DATABASE OPEN;
STBY:SQL> alter pluggable database MYPDB open;
STBY:SQL> show pdbs
--      CON_ID CON_NAME                                OPEN MODE  RESTRICTED
--  -----
--          2 PDB$SEED                                READ ONLY  NO
--          3 MYPDB                                    READ WRITE NO

-- ÜRETİM ORTAMINDA DROP EDİLEN TABLO, STANDBY VERİTABANINDA ERİSİLEBİLİR OLMUSTUR
STBY:MYPDB:SQL> select count(*) from system.deneme444;
--      COUNT(*)
--  -----
--          90939

-- KURTARILMASI İSTENEN TABLO DISARI EXPORT EDİLİR
-- EXPORT DOSYASI DAHA SONRA VERİ DONUSU İCİN KULLANILABİLİR
STBY:MYPDB:SQL> create directory ccebi_export as '/media/SharedFS/BACKUP/datapump';
[oracle@stbyvm ~]$ expdp 'userid=sys/welcome1@//localhost:1526/MYPDB as sysdba' \
FULL=N TABLES=SYSTEM.DENEME444 JOB_NAME=dummy_$(date +%Y%m%d_%H%M) \
DIRECTORY=CCEBI_EXPORT DUMPFILE=dummy_$(date +%Y%m%d_%H%M)_datapump.dmp \
LOGFILE=dummy_$(date +%Y%m%d_%H%M)_datapump.log logtime=ALL
-- ...
-- 23-MAY-15 12:11:18.284: Processing object type TABLE_EXPORT/TABLE/STATISTICS/MARKER
-- 23-MAY-15 12:11:24.154: . . exported "SYSTEM"."DENEME444"
10.37 MB   90939 rows
-- 23-MAY-15 12:11:24.457: Master table "SYS"."DUMMY_20150523_1211" successfully

```

loaded/unloaded

```
-- 23-MAY-15 12:11:24.463:
```

```
*****
```

```
-- 23-MAY-15 12:11:24.464: Dump file set for SYS.DUMMY_20150523_1211 is:
```

```
-- 23-MAY-15 12:11:24.469:
```

```
/media/SharedFS/BACKUP/datapump/dummy_20150523_1211_datapump.dmp
```

```
-- 23-MAY-15 12:11:24.490: Job "SYS"."DUMMY_20150523_1211" successfully completed at Sat
May 23 12:11:24 2015 elapsed 0 00:00:16
```

Gördüğünüz gibi kurguladığımız felaket senaryosundan alınımızın akıyla çıktık :-)

## 6. Standby Veritabanını Eski Hâline Alma

Tablomuzu kurtarmış olsak da, işimiz henüz bitmedi. Tabloyu kurtarmak için Standby veritabanını geçmişteki bir zamana çekip, read/write mode'a aldık. İşimiz tamamlandığına göre, standby veritabanı tekrar standby hâline gelebilir. Böylece her şeyi baştan kurmadan, veritabanı replikasyonuna kaldığımız yerden devam edebiliriz.

```
-- STANDBY VERITABANINI AKTIVASYON ONCESINE CEKIP
```

```
-- ACTIVE DATA GUARD OZELLIGINI ACIYORUZ
```

```
STBY:SQL> STARTUP MOUNT FORCE;
```

```
STBY:SQL> FLASHBACK DATABASE TO RESTORE POINT before_standby_activation;
```

```
STBY:SQL> ALTER DATABASE CONVERT TO PHYSICAL STANDBY;
```

```
STBY:SQL> STARTUP MOUNT FORCE;
```

```
STBY:SQL> ALTER DATABASE OPEN READ ONLY;
```

```
STBY:SQL> ALTER PLUGGABLE DATABASE MYPDB OPEN READ ONLY;
```

```
STBY:SQL> SHOW PDBS
```

```
-- CON_ID      CON_NAME                OPEN MODE  RESTRICTED
-----
-- 2           PDB$SEED                READ ONLY  NO
-- 3           MYPDB                   READ ONLY  NO
```

```
-- ARCHIVELOG DOSYALARINI UYGULAMAYA BASLIYORUZ
```

```
STBY:SQL> RECOVER MANAGED STANDBY DATABASE USING CURRENT LOGFILE DISCONNECT FROM SESSION;
```

```
STBY:SQL> select protection_mode, protection_level, database_role, switchover_status from
v$database;
```

```
-- PROTECTION_MODE      PROTECTION_LEVEL      DATABASE_ROLE      SWITCHOVER_STATUS
-----
```

```
-- MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PHYSICAL STANDBY NOT ALLOWED
```

```
/* STANDBY UZERINDEKI FLASHBACK KONTROLU */
```

```
SELECT NAME, OPEN_MODE, FLASHBACK_ON, LOG_MODE FROM V$DATABASE;
```

```
-- NAME      OPEN_MODE      FLASHBACK_ON      LOG_MODE
-----
```

```
-- CDBA      READ ONLY WITH APPLY YES      ARCHIVELOG
```

```
select name AS PARAMETER_NAME, value from v$parameter
```

```
where upper(name) in ('DB_FLASHBACK_RETENTION_TARGET', 'DB_RECOVERY_FILE_DEST_SIZE',
                     'DB_RECOVERY_FILE_DEST', 'LOG_ARCHIVE_DEST_1');
```

```
-- PARAMETER_NAME      VALUE
-----
```

```
-- log_archive_dest_1      LOCATION=/media/SharedFS/BACKUP/stby_archives
```

```
-- db_recovery_file_dest      /media/SharedFS/BACKUP/flashback_area
```

```
-- db_recovery_file_dest_size  21474836480
```

```
-- db_flashback_retention_target  600
```

```
-- BIRKAC ARCHIVELOG DOSYASI OLUSTURULUP, YANSIMASI KONTROL EDILIR
```

```
STBY:SQL> SELECT MAX(SEQUENCE#) FROM V$LOG_HISTORY;
```

```
-- STANDBY UZERINDE TRANSACTION LOG APPLY DURUMU KONTROL EDILIR
```

```
STBY:SQL> SELECT PROCESS, STATUS, SEQUENCE# FROM V$MANAGED_STANDBY;
```

```
-- PROCESS  STATUS      SEQUENCE#
-----
```

```
-- ARCH     CLOSING      100
```

```
-- ARCH     CLOSING      103
```

```
-- ARCH     CONNECTED     0
```

```
-- ARCH     CLOSING      102
```

```
-- RFS      IDLE         0
```

```
-- RFS          IDLE          104
-- RFS          IDLE          0
-- MRP0        APPLYING_LOG    104 --<< TRANSACTION LOG UYGULANIYOR
```

```
-- SON TEST OLARAK, 444 TABLOSUNUN OLMADIGI KONTROL EDILİYOR
```

```
-- 445 İSE PROBLEMSİZ ERİSİLEBİLİYOR
```

```
SQL> ALTER SESSION SET CONTAINER=MYPDB;
```

```
SQL> select count(*) from system.deneme444;
```

```
-- select count(*) from system.deneme444
```

```
--                               *
```

```
-- ERROR at line 1:
```

```
-- ORA-00942: table or view does not exist
```

```
SQL> select count(*) from system.deneme445;
```

```
-- COUNT(*)
```

```
-- -----
```

```
--      90940
```

Üretim ortamına dokunmadan zarar gören veriyi, standby üzerinden aldık. Daha sonra, standby veritabanını eski hâline getirdik. Oldukça severek kullandığım bu özelliği, en azından denemenizi tavsiye ederim. Zor ve stresli bir anınızda çok faydasını görebilirsiniz.

## 7. Kaynaklar

- *How to Implement a "snapshot standby" Using Oracle Data Guard 10 g Release 2*  
<http://www.oracle.com/au/products/database/dellprofile-128741.pdf>
- *Setup Flashback Database on Data Guard physical standby database for SAP customers*  
<http://www.oracle.com/us/solutions/sap/wp-ora4sap-flashback11g-1-303814.pdf>
- *11.2.0.2 Creating a Standby or a Clone Database using Rman Duplicate From Active Database*  
[https://blogs.oracle.com/AlejandroVargas/entry/11\\_2\\_0\\_2\\_creating](https://blogs.oracle.com/AlejandroVargas/entry/11_2_0_2_creating)