

ORACLE FLASHBACK DATABASE TEKNOLOJİSİ

Contents

1. FLASHBACK DATABASE Özelliđi	3
2. Veritabanını Hazırlamak	3
3. Flashback Database ile Recover	7
4. Flashback ile Zamanda İleri Geri Gitme	9
5. Flashback İin Önemli Kontroller.....	10
6. Flashback Recovery Area'nın Dolması.....	10
7. Kaynaklar	11

1. FLASHBACK DATABASE Özelliği

Flashback geniş bir konu. Drop edilen tabloların geri getirilmesinden, RMAN içinde kullanıma kadar birçok başlık bulunuyor. Konu çok geniş olduğundan bu yazıda sadece Flashback Database konusuna değineceğim.

Veri kayıplarını telafi etmek için çeşitli recover yöntemleri kullanılabilir. Fakat bu yöntemlerin en büyük eksikleri, işlem süresinin uzun oluşudur. Data kaybını sıfıra indirseniz bile, bozulan birkaç tabloyu geri getirmek için harcayacağınız süre, hatrı sayılır zaman alacaktır. Pratik, hızlı ve güvenli bir yöntem gerekmektedir. Yarım saat geriden gelen bir physical standby oluşturmak anlık hataların önüne geçmek için kullanılabilir bir yöntemdir. Fakat yetersiz kaldığı birçok nokta çıkacaktır. Oracle bu problemlerin önüne geçmek için FLASHBACK özelliğini sunmuştur.

FLASHBACK özelliği sayesinde, saatler sürecektir recover işlemini birkaç dakika içinde tamamlayıp; veritabanını geçmişteki bir zaman dilimine kolayca geri çekebilirsiniz. Hatta bununla kalmaz, veri tabanını geçmiş ve gelecek içinde getirip götürebilir ve read only mode'da sorgular çekebilirsiniz. Fiziksel hatalardan ziyade, özellikle kullanıcı tabanlı hatalar için bir can simidi olarak düşünebiliriz.

2. Veritabanını Hazırlamak

Flashback özelliğini kullanmak için üzerinden geçmemiz gereken belli başlı adımlar bulunuyor. Önce veritabanının flashback mode'da olup-olmadığını kontrol ediyoruz:

```
SQL> SELECT FLASHBACK_ON FROM V$DATABASE ;
FLASHBACK_ON
-----
NO
```

Görüldüğü gibi veritabanımızın flashback özelliği açık değil. Bunu değiştirmek amacıyla veritabanını kapatıp mount mode'da açmamız gerekiyor.

```
SQL> SHUTDOWN IMMEDIATE ;
SQL> STARTUP MOUNT ;
SQL> SELECT OPEN_MODE FROM V$DATABASE ;
OPEN_MODE
-----
MOUNTED
```

Flashback'ı aktif hâle getirmeden önce ikinci bir kontrol yapıyoruz. Flashback özelliği için veritabanı **ARCHIVELOG** şeklinde çalışmalıdır. Bunu da kontrol edip, gerekirse değiştiriyoruz:

```
SQL> SELECT LOG_MODE FROM V$DATABASE;

LOG_MODE
-----
NOARCHIVELOG
```

Sorgu sonucuna bakarak veritabanının arşiv modda olmadığını görüyoruz. Gerçek işlemlerin yürütüldüğü, veri kaybına toleransın gösterilemeyeceği hiçbir ortamda NOARCHIVELOG şeklinde çalışmaz. Ancak örnek olarak kullandığım veritabanı, test amaçlı olduğundan şimdiye kadar ARCHIVELOG'a çekme ihtiyacı duyulmamış. Şimdi ise flashback'in kullanılabilmesi için veritabanının arşiv mode'a geçmesini sağlamamız gerekiyor!

Önce arşiv dosyalarının yazılacağı hedef klasörü **"/data2/TESTDB/archive/"** şeklinde ayarlıyoruz:

```
SQL> ALTER SYSTEM SET log_archive_dest_1='LOCATION=/data2/TESTDB/archive/';
```

Kontrol edelim:

```
SQL> show parameter log_archive_dest_1

log_archive_dest_1      string      LOCATION=/data2/TESTDB/archive/
```

Dosyaların yazılacağı lokasyon bilgisi tutuyorsa; archivelog mode desteğini açalım ve log_mode bilgisini kontrol edelim:

```
SQL> ALTER DATABASE ARCHIVELOG;

Database altered.

SQL> SELECT LOG_MODE FROM V$DATABASE;

LOG_MODE
-----
ARCHIVELOG
```

Her şey tamamsa flashback için parametreleri düzenleyebiliriz.

DB_FLASHBACK_RETENTION_TARGET, **DB_RECOVERY_FILE_DEST_SIZE** ve **DB_RECOVERY_FILE_DEST** olmak üzere bizi ilgilendiren üç temel parametre vardır. Bu parametreleri kısaca özetlersek:

- **DB_FLASHBACK_RETENTION_TARGET**: Veritabanını kaç dakika geriye alabileceğinizi gösterir. Örneğin 600 değerini girerseniz, 10 saatlik zaman diliminde herhangi bir noktaya gidebilirsiniz.

- **DB_RECOVERY_FILE_DEST_SIZE** : Yedek işlemleri için ayrılan alanın boyutunu belirler. 200 GB'lik bir disk alanına sahip olabilirsiniz. Ama 20 GB'tan fazlasını tahsis etmek size cazip gelmeyebilir. Bu sebeple parametre değerini, DB_RECOVERY_FILE_DEST_SIZE=20G şeklinde ayarlayıp, n'olursa olsun 20GB'i geçme diye belirtmeniz mümkündür.

Fakat bu parametre bir üstte gördüğümüz **DB_FLASHBACK_RETENTION_TARGET** değeri ile çakışabilir. Örneğin 10080 dakika (yani 7 gün) boyunca herhangi bir zamana dönebilecek şekilde **DB_FLASHBACK_RETENTION_TARGET** parametresini ayarlayabilirsiniz. Yalnız DB_RECOVERY_FILE_DEST_SIZE, 20G şeklinde ayarlıysa ve siz sadece bir günde 20G'in üstünde flashback verisi yarattıysanız, girilen 10080 değerinin pek bir anlamı kalmayacaktır.

ÖNEMLİ BİR NOT: 200 GB'lik bir diskiniz varsa ve burayı Flashback Recovery Area için kullanma niyetindeyseniz; diskinizin tamamını bu işe ayırmak akıllıca olmayacaktır. Maksimum %90 kadar bir disk alanını (örneğimiz için 180 GB) tahsis etmek daha uygundur. Flashback Recovery Area'nın dolduğuna dair uyarılar geldiğinde, panik yapmadan ve veritabanını kapatmaya gerek duymadan çözüm üretmeniz mümkün olacaktır. Müsait bir zamanda da bu duruma çözüm üretebilirsiniz.

- **DB_RECOVERY_FILE_DEST** : Flashback bilgisinin nerede tutulacağını belirtir. Arşiv dosyalarının koyulacağı lokasyonu belirten **log_archive_dest_1** parametresine benzetebiliriz.

Bu parametrelere göre ayarlamalarımızı yapalım. Veritabanı için Flashback recovery area'yı maksimum 20GB olsun. Flash recovery area lokasyonu olarak **/data2/TESTDB/flashback_area/** klasörünü verelim. Geri dönebileceğimiz zaman olaraksa 600 dakikalık (10 saat) bir aralık ayaralayalım.

```
-- 10 SAAT (600 DAKIKA) FLASHBACK AYARI
SQL> ALTER SYSTEM SET DB_FLASHBACK_RETENTION_TARGET=600;
System altered.
-- 20GB UST LIMIT
SQL> ALTER SYSTEM SET DB_RECOVERY_FILE_DEST_SIZE=20G;
System altered.
```

```
-- LOKASYON BELIRLENIYOR
SQL> ALTER SYSTEM SET
      DB_RECOVERY_FILE_DEST='/data2/TESTDB/flashback_area/';
System altered.
```

Bazı durumlarda aşağıdaki gibi bir hata mesajıyla karşılaşabilirsiniz:

```
ERROR at line 1:
ORA-02097: parameter cannot be modified because specified value is invalid
ORA-16018: cannot use LOG_ARCHIVE_DEST with LOG_ARCHIVE_DEST_n or
DB_RECOVERY_FILE_DEST
```

Aşağıdaki adımları takip ederek, sorunu çözebilirsiniz:

```
-- ARSIV LOKASYONUNU SIFIRLAYIN
SQL> ALTER SYSTEM SET log_archive_dest='';

-- FLASHBACK ICIN KULLANILACAK ADRESI TEKRAR GIRIN
SQL> ALTER SYSTEM SET
      DB_RECOVERY_FILE_DEST='/data2/TESTDB/flashback_area/';

-- ARSIV LOKASYONUNU TEKRAR GIRIN
SQL> ALTER SYSTEM SET
      log_archive_dest_1='LOCATION=/data2/TESTDB/archive/';
```

Gerekli değişiklikleri yaptıktan sonra, aşağıdaki ifadeyi çalıştırıp, ayarları kontrol edebilirsiniz:

```
SET FEEDBACK OFF
SET LINES 130
COLUMN NAME FORMAT A30
COLUMN FLASHBACK_ON FORMAT A30
COLUMN LOG_MODE FORMAT A50
SELECT NAME, FLASHBACK_ON, LOG_MODE FROM V$DATABASE;
COLUMN NAME FORMAT A30
COLUMN VALUE FORMAT A80
select name,value from v$parameter
where upper(name) in
('DB_FLASHBACK_RETENTION_TARGET', 'DB_RECOVERY_FILE_DEST_SIZE',
 'DB_RECOVERY_FILE_DEST', 'LOG_ARCHIVE_DEST_1');
```

NAME	FLASHBACK_ON	LOG_MODE
YATIRIM	NO	NOARCHIVELOG

NAME	VALUE
log_archive_dest_1	LOCATION=/data2/TESTDB/archive/
db_recovery_file_dest	/data2/TESTDB/flashback_area/
db_recovery_file_dest_size	21474836480
db_flashback_retention_target	600

Bu noktaya kadar her şey doğruysa, flashback özelliğini aktif hâle getirip, veritabanını açabiliriz:

```
-- FLASHBACK OZELLIGINI AKTIF HALE GETIRELİM
SQL> ALTER DATABASE FLASHBACK ON;

Database altered.
```

```
-- VERITABANINI ACALIM
SQL> ALTER DATABASE OPEN;

SQL> SELECT OPEN_MODE FROM V$DATABASE;

OPEN_MODE
-----
READ WRITE
```

3. Flashback Database ile Recover

Hazırlık aşamasını anlatmam uzun sürse de, recover işlemi birkaç komuttan ibaret. Önemli olan hangi zamana dönülmesi gerektiğini bulmak. İki şekilde recover işlemi gerçekleştirmemiz mümkündür. Bunlardan birisi, SCN – System Change Number üzerinden giderek yapılır. SCN, veritabanında yapılan her işlem için sürekli olarak artan nümerik bir değerdir. Mevcut SCN değerini aşağıdaki sorguyla öğrenebilirsiniz:

```
SQL> SELECT CURRENT_SCN FROM V$DATABASE;

34530329701
```

SCN değerinin hangi zamana; ya da zamanın hangi SCN değerine denk düştüğünü anlayabilmek için iki adet fonksiyon bulunmaktadır:

```
-- SCN DEGERINI ZAMANA CEVIRIYORUZ
SQL> SELECT SCN_TO_TIMESTAMP( 34530329701 ) FROM DUAL;

24.10.2008 16:46:55,000000000
```

```
-- ZAMANI SCN DEGERINE CEVIRIYORUZ
SQL> SELECT TIMESTAMP_TO_SCN(
        TO_TIMESTAMP('2008-10-24 16:46:00', 'YYYY-MM-DD HH24:MI:SS') )
FROM DUAL;
```

Bu yapıda ihmal edilebilecek kadar ufak tefek farklar çıkabilir. Örneğin aşağıdaki sorgunun sonucu kendisiyle aynı çıkması gerekirken, ihmal edilebilecek çok çok ufak bir fark çıkmıştır:

```
SQL> SELECT TIMESTAMP_TO_SCN( SCN_TO_TIMESTAMP( 34530329701 ) ) FROM DUAL;

34530329700
```

SCN değeri her veritabanı için kendine özgüdür. Ve bir süre sonra geçerliliğini yitirir. Örneğin 1 ay önceki SCN değerini öğrenemezsiniz.

Flashback ile recover ederken kullanabileceğiniz ikinci bir yöntemse direkt tarih vermektedir. Gerçi elinizde tarih bilgisi olduktan sonra, SCN değerini alabilir ya da SCN ile zamanı belirleyebileceğinizden söz etmiştik. Bu iki fonksiyon elinizde olduktan sonra zaman kullanarak dönmeniz çok da şart değildir.

Şöyle bir durum düşünelim: Kritik bir işlem yapacağız ve beklediğimiz sonuç gelmezse geri almamız gerekiyor. Ancak işlem o kadar teferruatlı ki, basit bir rollback ile kurtulmanız mümkün olmuyor. Tablolar değişiyor; yeni prosedürler yaratılıyor; eski paketler drop ediliyor, yenileri yaratılıyor vs... İşlemin başarısız olduğunu gördüğünüzde, birkaç dakika içinde geriye dönmek hoş olmaz mıydı? Bildiğimiz recover işlemiyle bunu yapmak saatler sürecektir. Hâlbuki Flashback ile dakikalar içinde bu dönüşü gerçekleştirebiliriz. Önemli olan işleme başlamadan önce SCN değerini almaktır.

İşleme başlamadan önce saate baktık ve **16:46:55** olarak not ettik. SCN değerini ise **34570271536** olarak kaydettik.

```
SQL> SELECT CURRENT_SCN, SYSDATE FROM V$DATABASE;
```

```
CURRENT_SCN    SYSDATE
-----
34570271536    24.10.2008 16:46:55
```

Problem çıktı ve veritabanını eski hâline döndürmemiz istendi. Yapmamız gereken veritabanını kapatıp, mount mode'da başlatmak ve ardından flashback özelliğinden yararlanmak olacak:

```
SQL> SHUTDOWN IMMEDIATE
SQL> STARTUP MOUNT
SQL> FLASHBACK DATABASE TO SCN 34570271536;
Flashback complete.
```

Şimdi de veritabanını açmamız gerekiyor:

```
SQL> ALTER DATABASE OPEN RESETLOGS;
```


Burada dikkat etmemiz gereken en önemli konu, yapılan işlemin veritabanına ait incarnation number'ı değiştireceğidir. Bu durumdan standby veritabanlarınız da etkilenecektir. Pek arzu edeceğiniz bir durum değil; fakat ana ortamda data kaybı (veya bozulması) olmayacaktır.

Verdiğimiz örnekte, SCN üzerinden recover ettik. Zaman kullanarak da recover etmeniz mümkündür. Aşağıdaki gibi tarih bilgisiyle de bu işi yapabiliydik:

```
SQL> FLASHBACK DATABASE TO TIMESTAMP
      to_timestamp('24.10.2008 16:46:55', 'YYYY-MM-DD HH24:MI:SS');
Flashback complete.
```

4. Flashback ile Zamanda İleri Geri Gitme

Bazı durumlarda bozulmanın ne zaman olduğunu bilemeyiz. Bizden bozulma anının hemen öncesi istenebilir. Böyle bir durumda, flashback özelliğini adım adım kullanmamız ve database'i read only mode'a çekerek kontrol yapmamız mümkündür.

Recover adımları aynıdır. Yine tarih bilgisi veya SCN ile recover işlemi gerçekleştiririz. Fakat bu sefer open resetlogs yapmak yerine, veritabanını read only mode'da açarız:

```
SQL> FLASHBACK DATABASE TO TIMESTAMP
      to_timestamp('24.10.2008 16:36:55', 'YYYY-MM-DD HH24:MI:SS');
Flashback complete.
```

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

Doğru zamanda değilseniz, veritabanını kapatıp, yukarıdaki işlemi farklı bir zaman dilimi vererek tekrarlayabilirsiniz. Uygun zamanı bulduğunuzda ise aşağıdaki komutu vererek veritabanını kullanıma hazır hâle getirmeniz mümkündür:

```
SQL> ALTER DATABASE OPEN RESETLOGS;
```

Peki bu işlemten vazgeçersek ya da son ana geri dönmek istersen ne yapacağız? O zaman aşağıdaki komutu kullanmanız yeterlidir:

```
SQL> RECOVER DATABASE;
```

Diğer bütün flashback komutları incarnation değerini değiştirir. Ama son aşamada verdiğimiz komutta veritabanı incarnation değeri değişmeyecektir.

5. Flashback İçin Önemli Kontroller

Flashback'i efektif kullanmak için bazı sorguları kullanabiliriz. Örneğin ne kadar eskiye gidebileceğimiz ya da Flashback recovery area'nın ne kadarının dolduğunu görmek işimize hâyli yarayacaktır.

```
/* FLASHBACK LOG'LARI GOSTERIR */
SELECT * FROM V$FLASHBACK_DATABASE_LOGFILE;

/* GIDILEBİLECEK EN ESKE FLASHBACK ZAMANINI GOSTERIR */
SELECT * FROM V$FLASHBACK_DATABASE_LOG;

/* BUTUN TABLESPACE'LER ICIN FLASHBACK ACIK OLMALIDIR */
SELECT NAME TBS_NAME, FILE_NAME DB_FILE, FLASHBACK_ON FB_LOGGING,
       ONLINE_STATUS STATUS
FROM   V$TABLESPACE, DBA_DATA_FILES WHERE NAME=TABLESPACE_NAME;

/* FLASHBACK RECOVERY AREA DOLULUK ORANINI GOSTERIR */
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;
```

6. Flashback Recovery Area'nın Dolması

Yoğun işlemin söz konusu olduğu bir veritabanında flashback recovery area için ayrılan alanın (*DB_RECOVERY_FILE_DEST_SIZE*) dolması mümkündür. Ancak belirlenen alanın üstüne çıkılmayacaktır. 60 GB ayrılan alan, 60 GB'i geçemez. Peki bu durumda ne olur?

Flashback log dosyaları (*flb*), sirküler bir yapıdadır. Oracle Flashback recovery area için alanın yetersiz olduğuna kanaat getirirse, eski flashback log dosyalarını tekrar kullanıma alır. Denemelerimde gördüğüm kadarıyla bunu başarıyla yapıyor. Fakat sıkıntısı, işlemin daha yavaş gerçekleşmesi ve geriye dönebileceğiniz zamanı kısmasıdır. Eski log dosyaları tekrar kullanılacağından, arzu ettiğiniz geri dönüş zamanını yakalayamayabilirsiniz.

Performans konusuna gelince... Flashback recovery area hemen dolacak kadar küçük verilmemeli. Çok ufak verilirse, hızlı bir şekilde bu alan dolacak ve eski dosyaların tekrar tekrar kullanılmasından problem çıkacaktır. Şayet alan sıkıntınız varsa ve yeni alan tahsis

edemiyorsanız, yapabileceğiniz en uygun şey flashback log dosyalarını ara ara temizlemektir. Bunu elle, işletim sistemi üzerinden yapmak uygun değildir. Oracle dökümanlarında RMAN vb. araçlar kullanılarak yapılması öneriliyor.

Benim bulduğum bir yöntem ise, flashback özelliği kapatıp, sonra tekrar açmak şeklinde. Bunu yaptığınızda, RVWR - Recovery Writer process'i durmakta ve flashback'e ait olan bütün log'lar Oracle tarafından silinmektedir. Bu konudaki tek handikap, veritabanını kapatmanın gerekliliğidir.

```
SQL> SHUTDOWN IMMEDIATE;
SQL> STARTUP MOUNT;
SQL> ALTER DATABASE FLASHBACK OFF;

/*
##### BU ESNADA ALERT LOG DOSYASI #####
...
ALTER DATABASE FLASHBACK OFF
Fri Oct 24 12:15:50 2008
Stopping background process RVWR
Deleted Oracle managed file
/data2/TESTDB/flashback_area/o1_mf_4j2zm7z5_.flb
Deleted Oracle managed file
/data2/TESTDB/flashback_area/o1_mf_4j316ybk_.flb
...
Flashback Database Disabled
Completed: ALTER DATABASE FLASHBACK OFF
*/

SQL> ALTER DATABASE FLASHBACK ON;
SQL> ALTER DATABASE OPEN;
```

Bunu işlemi pek sık yapmanız gerekmiyor. Hatta hiç yapmanız gerekmiyor. Fakat biraz daha performanslı çalışmak ya da disk alanı konusunda sıkıntı yaşadığınızda ciddi oranda tasarruf sağlamak istiyorsanız, uygulayabilirsiniz.

7. Kaynaklar

http://www.psoug.org/reference/flash_db.html

http://users.telenet.be/oraguy.be/flashback_database.htm

<http://arjudba.blogspot.com/2008/04/ora-16018-and-ora-16019-logarchivedest.html>

<http://appsdbanew.wordpress.com/2008/06/06/enabling-flashback-on-oracle/>

http://download.oracle.com/docs/cd/B28359_01/backup.111/b28270/rcmflash.htm

http://www.dba-oracle.com/p_db_flashback_recovery_target.htm