

Tibero Backup & Recovery

Contents

- 0. 교육소개
- 1. 백업 및 복구
- 2. 백업 및 복구 예제 1
- 3. 백업 및 복구 예제 2



0.

교육소개

과정	내용
TIBERO Backup & Recovery	<p>Tibero 데이터베이스 백업, 복구를 학습합니다.</p> <ul style="list-style-type: none">• 교육기간 : 2일• 교육대상 : Tibero 관리자



1. 백업 및 복구

Introduction

● Backup 및 recovery Introduction

- 여러 가지 유형의 장애로부터 데이터베이스를 보호
 - ✓ MTBF (Mean Time Between Failure)를 증가시키고 MTTR (Mean Time Between Recover)를 감소
- 시스템 장애 발생시 복원을 하거나, 시스템 작동을 유지하기 위한 절차 또는 기법
- 관리자는 시스템 장애시 발생할 손실을 최소화하고 복구 가능한 상태로 데이터베이스를 운용해야 함.
 - ✓ 최소한 한 달에 한번 데이터베이스 전체 백업
 - ✓ 하루에 한번씩 Export 백업 권장
- 데이터베이스 관리자는 백업에 대한 정책을 수립하고 꼭 필요한 데이터를 최소한의 양으로 백업
 - ✓ 백업은 DBA의 주요 역할 중 가장 주의를 기울여야 함
- “RECOVERY에 실패한 DBA는 용서할 수 있지만, BACKUP을 실패한 DBA는 용서할 수 없다.”
- 백업이 정상적으로 수행되었는지 주기적으로 검증하는 것을 권장한다.

여러 가지 유형의 장애

- 명령문의 실패
- 사용자 프로세스의 실패
- 사용자로 인한 장애
- Instance fail
- Media fail

여러 가지 유형의 장애

● 명령문의 실패의 원인

- 테이블의 제약 조건에 위배되는 데이터를 insert
- 권한의 부족
- 테이블 생성시나 데이터 변경시 테이블스페이스의 공간이 부족한 경우

여러 가지 유형의 장애

● 사용자 프로세스의 실패 및 해결

- 비정상적인 종료로 인한 사용자 프로세스의 실패가 대부분
- Tibero의 monitor process가 비정상적인 종료를 감지하고 수행중인 트랜잭션 등은 롤백

● 사용자로 인한 장애 및 해결

-장애 발생 상황

DROP TABLE

TRUNCATE TABLE

DELETE FROM & COMMIT

잘못된 update & COMMIT

-해결 방안

사용자에 대한 교육 실시

백업에서 복구

Export 받은 파일에서 테이블을 import

Time-based 불완전 복구

여러 가지 유형의 장애

● Instance fail

- 정전, CPU나 memory fault, background process의 비정상적인 종료 등 대부분

- 복구

 - 특별한 복구 작업이 필요하지 않음

 - tbboot을 통해 DBMS를 기동하면 자동으로 복구됨

 - 로그 등을 참고하여 원인 분석

● Media fail

- 데이터파일이 저장된 하드 디스크의 장애

- 데이터 파일의 삭제

- 정해진 복구 전략에 따라 복구 절차가 필요

Backup 및 Recovery의 전략 수립

- 업무적인 요구 사항

- MTBF (Mean Time Between Failure)를 증가시키고 MTTR (Mean Time To Recover)를 감소

- 운영 요구 사항

- 24*365 운영, 백업 테스트 및 검증, 데이터베이스의 변경, 백업 데이터의 보관 장소 등

- 기술적 고려 사항

- 하드웨어, 소프트웨어의 구성, 백업 주기 결정을 위한 트랜잭션 주기, 데이터의 양 등

- 경영진 합의

- 경영진에서 기대하는 시스템의 가용성, 백업 및 복구 절차에 대한 이해, 백업을 위한 리소스 확보 등

Backup과 Recovery 관련 Tibero 동작 방식의 이해

● Transaction Durability

- Committed Transaction MUST Survive failures (Recoverable)

● Logging

- Redo Log Buffer : TSM에 redo 데이터를 저장하기 위한 영역
- Redo Log File : recovery를 위해 가장 중요한 파일
- Archive : archive log mode시 redo logfile을 별도의 위치에 backup

● 관련된 background process

- DBWR, RCWP (tibero 6 FS07 ~)
- DBWR, RECO (tibero 6 ~ tibero 6 FS06)
- LGWR, DBWR, CKPT, LARC (~ tibero 5)

Backup과 Recovery 관련 Tibero 동작 방식의 이해

● Redo 저장 대상 범위

–Physical Logging

- 수정이 일어날 때마다 해당 block을 통째로 남긴다.
- 바뀔 때마다 block size 만큼의 로그가 남아 많은 양의 유지해야 한다.

–Logical Logging

- update, delete 같은 operation을 log에 남기는 방법이다.
- 여러 physical block들에 대한 수정을 하더라도 operation만 기록되어 logging 양이 적다.
- 하지만, change가 반드시 순서대로 apply되어야 한다. recovery 때 log apply가 어렵다.

–Physiological Logging

- physical logging과 logical logging의 장점을 합친 형태
- block의 physical change를 기록하는 change vector 들로 구성된 redo record 들로 이루어진다.
- change vector는 atomic block change 이며, redo record는 atomic database change

Backup과 Recovery 관련 Tibero 동작 방식의 이해

● Redo 저장 시 일어나는 일들

–Page fix rule

- block에 대한 수정작업이 일어나기 전에 buffer에 대한 Lock을 잡는다.
- 실제 data buffer를 수정하기 전에 Redo Record를 Generation한다.

–WAL (Write Ahead Logging)

- database buffer가 변경되기 전에 redo record 가 먼저 log buffer에 기록된다.
- DBWR가 dirty buffer를 disk에 write하기 전에 Redo Record가 먼저 log buffer의 필요한 redo record를 log file에 먼저 기록한다.

–Log force at commit

- Commit 발생시 log buffer의 redo entry를 모두 log file에 write한다.
- Commit된 데이터의 보장.

–Online log switch

- Online Log중 다음의 조건을 만족하는 Log file로 Switching 된다.

Not active

Check point complete

Archiving not required

Backup과 Recovery 관련 Tibero 동작 방식의 이해

● DATABASE (Controlfile, Redolog file, Datafile) 동기화 방식

-TSN

- Database의 version 또는 commit version
- Data Concurrency Control, Redo Ordering, Recovery 등에 사용된다.
- Transaction이 commit될 때 TSN이 generate된다.

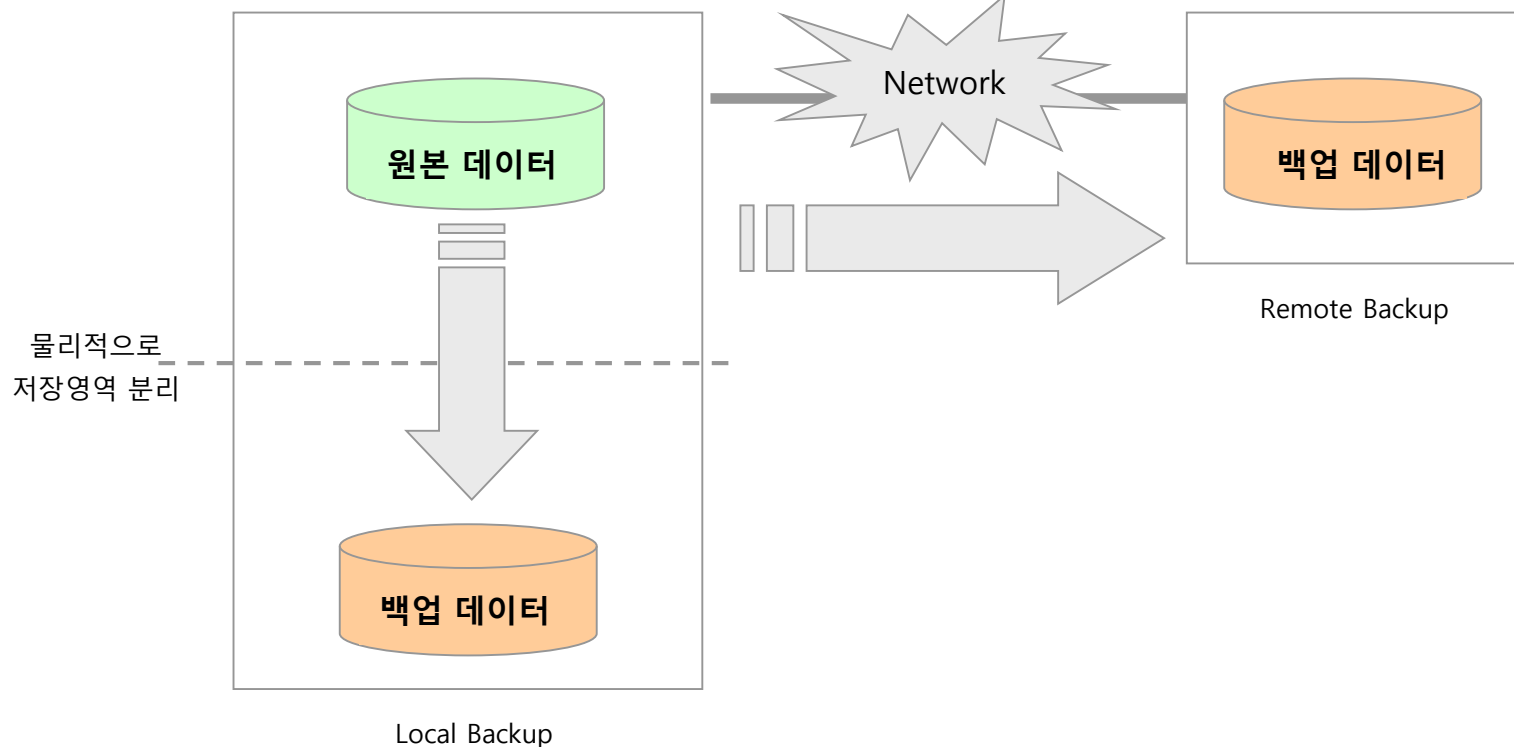
-Checkpoint

- 주기적으로 혹은 사용자의 을 줄임. 요청에 따라 메모리에 있는 모든 변경된 (dirty) 블록을 디스크에 쓰는 작업
 - 복구에 필요한 logfile의 양
 - CheckPoint 발생 상황
 - 모든 로그 스위치 발생시
 - 인스턴스가 NORMAL, POST_TX, IMMEDIATE 옵션으로 종료
 - 사용자 요청에 따라 수동으로 발생
- ```
SQL> ALTER SYSTEM CHECKPOINT;
```
- ALTER TABLESPACE [BEGIN BACKUP | END BACKUP]
  - Checkpoint 발생 시, DBWR가 작동하기 전에 먼저 LGWR이 현재 log buffer의 내용을 online log file에 write하고 해당 dirty buffer에 mark가 되면, 이 정보를 DBWR이 받아서 모든 marked dirty buffer를 disk에 기록한다.
  - Checkpoint는 Checkpoint TSN 이전에 발생한 Online log file내의 모든 변경 사항이 Disk에 저장되었음을 의미한다.

# Backup 개요

## ● Backup 개요

- 시스템 장애 발생시 복원을 하거나, 시스템 작동을 유지하기 위한 절차 또는 기법
- 관리자는 시스템 장애시 발생할 손실을 최소화하고 복구 가능한 상태로 데이터베이스를 운용해야 함.  
최소한 한달에 한번 데이터베이스 전체 백업  
하루에 한번씩 Export 백업 권장
- 데이터베이스 관리자는 백업에 대한 정책을 수립하고 꼭 필요한 데이터를 최소한의 양으로 백업





# Backup 종류와 전략

- Backup의 종류

- 논리적인 백업

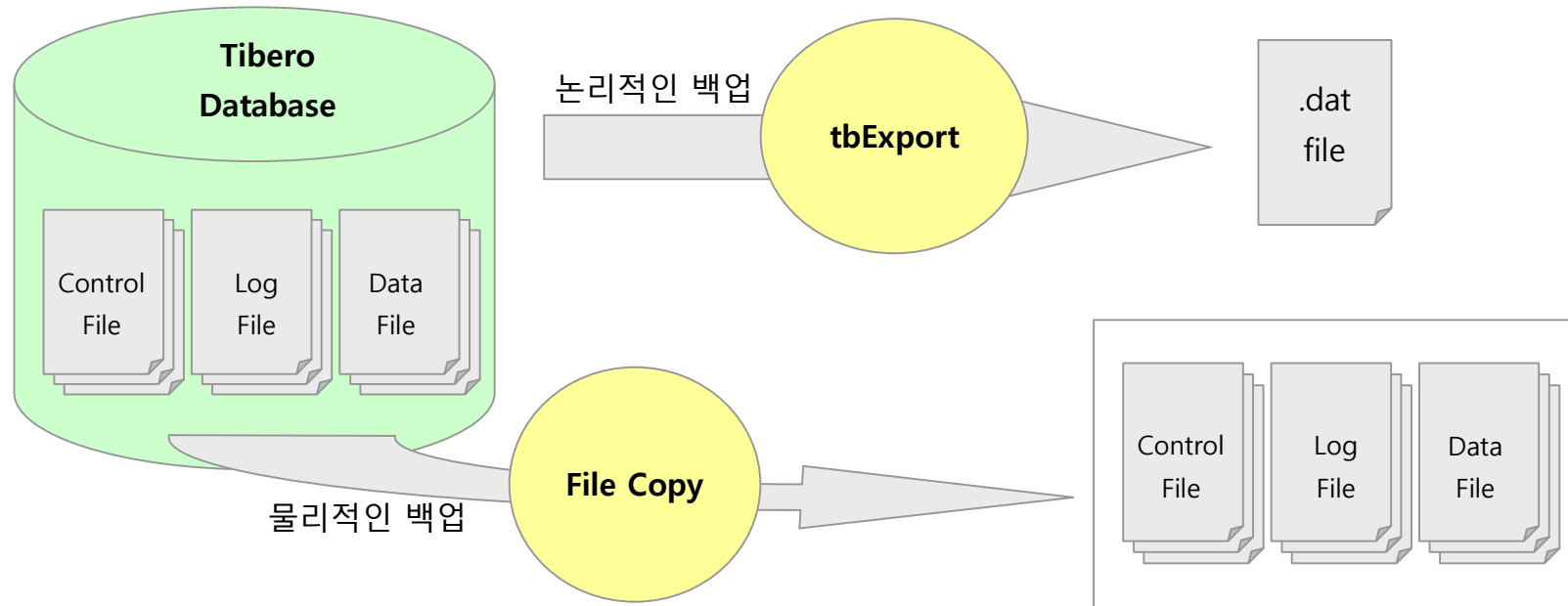
- 데이터베이스의 논리적인 단위 백업.

- Table, Index, Constraint, Sequence 등...

- Export 툴로 백업

- 물리적인 백업

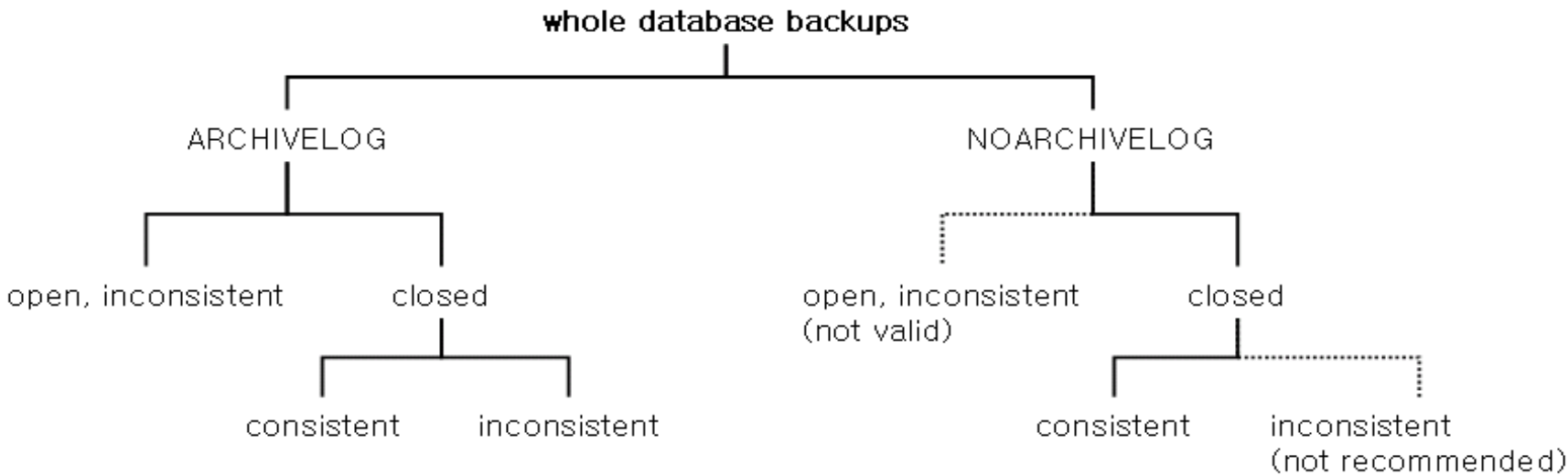
- 데이터베이스를 구성하는 파일을 운영체제 레벨에서 COPY 명령으로 백업 datafile, controlfile, archive logfile



# Backup 종류와 전략

## ● Backup의 종류

| 구분                                                      | 설명                                                                                                                    |
|---------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|
| NOARCHIVELOG 모드에서의 백업<br>(Offline Backup / Cold Backup) | 데이터베이스를 구성하는 전체 파일에 대해 운영을 멈춘 상태에서 백업<br>데이터베이스를 백업 받은 시점으로의 복구만 가능                                                   |
| ARCHIVELOG 모드에서의 백업<br>(Online Backup / Hot Backup)     | 운영중에도 백업가능. Controlfile 생성문, Datafile, Archive logfile 백업<br>백업된 archive logfile 의 시점에 따라 datafile 백업 시점 이전으로의 복구도 가능 |



| 구분              | 설명                                          |
|-----------------|---------------------------------------------|
| Consistent 백업   | - 정상적인 Shutdown 후의 Backup                   |
| Inconsistent 백업 | - DB 운영중에 Backup 또는 정상종료 되지 않은 상태에서의 Backup |

# Backup 대상 - Controlfile & Redo logfile

## ● Controlfile 기능

- 데이터베이스의 구조를 이진 파일 형태로 저장
- 데이터베이스를 mount할 때 반드시 필요
- 파일이 없으면 복구를 하거나 재생성해야 한다.
- 두 개 이상의 controlfile로 구성하는 것을 권장하고 서로 다른 디스크에 위치시키도록 한다.

## ● 동적뷰

- v\$controlfile

## ● 다중화 방법

- 데이터베이스를 down (tbdown)
- Control file을 다른 위치로 copy
- \$TB\_SID.tip 파일에서 CONTROL\_FILES 파라미터에 추가
- 데이터베이스를 기동 (tbboot)

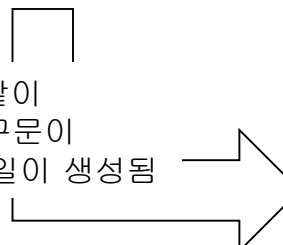
# Backup 대상 - Controlfile & Redo logfile

## ● Controlfile 백업

- Offline backup : O/S의 copy 명령을 통해 별도의 위치에 copy한다.
- Online backup : 다음과 같이 controlfile 구문을 생성하여 copy 한다.

```
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE
AS [백업할 파일 경로 및 이름] REUSE NORESETLOGS;
```

예) 다음과 같이  
Controlfile 구문이  
들어있는 파일이 생성됨



```
CREATE CONTROLFILE REUSE DATABASE "tibero"
LOGFILE
 GROUP 0 ('/home/tibero6/tbdata/log01.log', '/home/tibero6/tbdata/log02.log') SIZE 3M,
 GROUP 1 ('/home/tibero6/tbdata/log11.log', '/home/tibero6/tbdata/log12.log') SIZE 3M,
 GROUP 2 ('/home/tibero6/tbdata/log21.log', '/home/tibero6/tbdata/log22.log') SIZE 3M
NORESETLOGS
DATAFILE
 '/home/tibero6/tbdata/system001.dtf',
 '/home/tibero6/tbdata/undo001.dtf'
NOARCHIVELOG
MAXLOGFILES 100
MAXLOGMEMBERS 8
MAXDATAFILES 256;
```

# Backup 대상 - Controlfile & Redo logfile

## ● Redo logfile이란?

- 데이터베이스의 모든 변경 사항을 저장한다.
- 최소한 2개 이상의 redo log group으로 구성
- 각 log group은 하나 이상의 redo logfile로 구성

## ● 동적뷰

- V\$LOG, V\$LOGFILE

## ● 다중화

- Logfile의 손실을 대비하여 각각의 redo log group은 2개 이상의 logfile로 구성할 것을 권장
- Log group의 logfile들은 별도의 disk로 분리하도록 한다.
- Log group의 추가

```
ALTER DATABASE ADD LOGFILE
```

```
GROUP 4 ('/home/tibero6/tbdata/redo41.log', '/home/tibero6/tbdata/redo42.log') SIZE 10m;
```

- Log group에 member 추가

```
ALTER DATABASE ADD LOGFILE MEMBER '/home/tibero6/tbdata/redo43.log' TO GROUP 4;
```

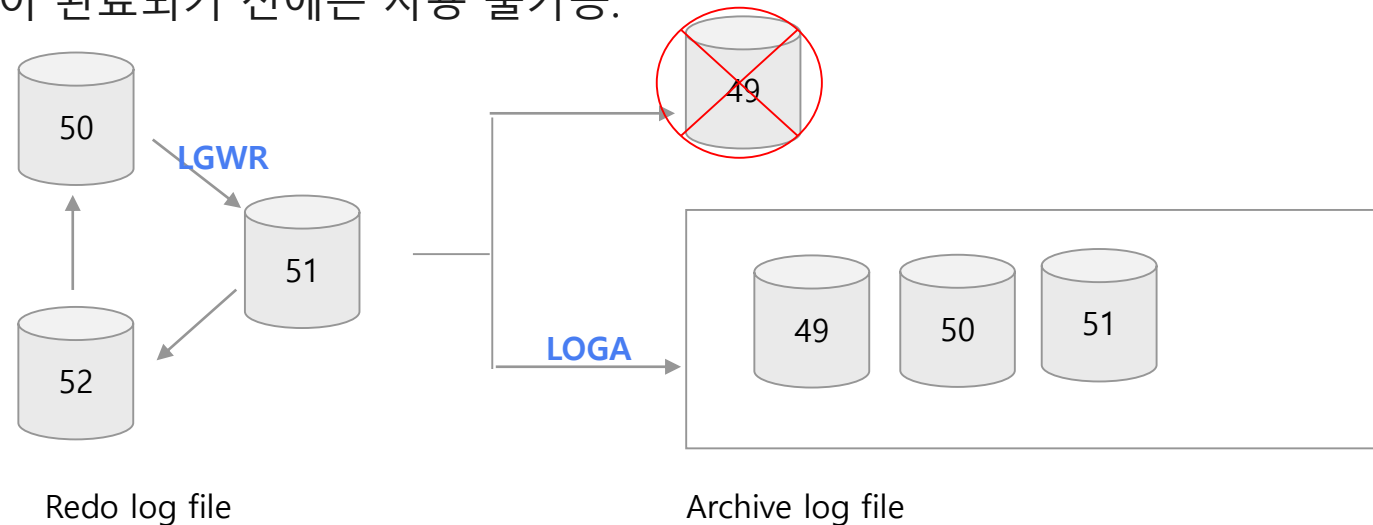
# Backup 방식의 차이 - NOARCHIVELOG Mode VS ARCHIVELOG mode

## ● NOARCHIVELOG Mode

- 기본적으로 데이터베이스는 NOARCHIVELOG Mode.
- Redo logfile은 순환하여 사용되고 log switch가 발생하면 이전의 logfile을 overwrite.  
이전의 log가 없기 때문에 트랜잭션 기록 중 최근의 부분만 사용할 수 있다
- Redo log가 overwrite되면 결국 마지막 전체 백업(offline backup)을 통한 복구만 가능.
- 백업은 반드시 데이터베이스가 정상 종료된 상태에서 해야 하며 이로 인해 서비스의 중지가 발생.

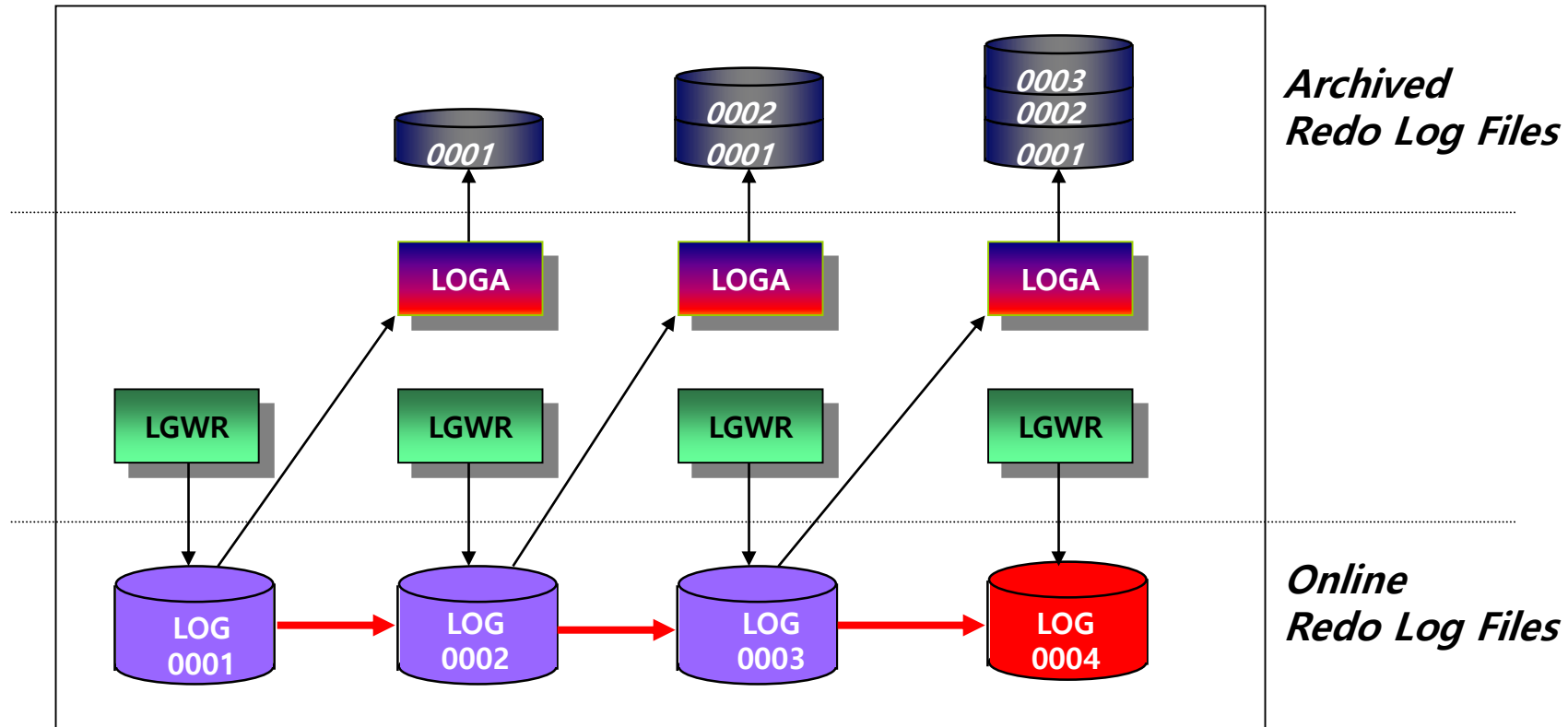
## ● ARCHIVELOG Mode

- Redo logfile은 LOGA(Log Archiver)에 의해 백업이 완료되기 전에는 사용 불가능.
- Archive logfile은 media recovery에 사용 가능.
- 데이터베이스가 기동 중에 백업 가능.



# Backup 방식의 차이 - NOARCHIVELOG Mode VS ARCHIVELOG mode - Archive log mode

- Archive log mode에서 On-Line Redo Log 파일 사용



# Backup 방식의 차이 - NOARCHIVELOG Mode VS ARCHIVELOG mode - Archive log mode

## ● Archive log mode란?

- Log switch시 Redo logfile을 별도의 분리된 위치로 복사하여 저장하여 운영하는 방법
- Archive logfile은 recovery시 사용되는 매우 중요한 파일이다.
- Online backup을 하려면 반드시 archive log mode로 운영해야 한다.

## ● 관련 background process / initial parameter

- LOGA : tiberio 기동시 자동으로 종료된다.
- LOG\_ARCHIVE\_DEST : archive logfile이 저장되는 위치

## ● 변경 방법

- 데이터베이스 down
- 데이터베이스를 mount 모드까지 기동  
\$tbboot mount
- SYS 유저로 접속하여 archive log mode로 변경  
\$tbsql sys/tibero  
SQL> ALTER DATABASE ARCHIVELOG;
- 데이터베이스 종료 후 재기동

## ● Archive log mode 확인

- V\$DATABASE : LOG\_MODE 컬럼을 조회한다.



# Backup 방식의 차이 - NOARCHIVELOG Mode VS ARCHIVELOG mode - Archive log mode

- Archive log 정보

- 아카이브 대상 상태 이해

- V\$ARCHIVE\_DEST\_FILES
      - LOG\_ARCHIVE\_DEST내의 archive log files의 정보 표시
    - V\$ARCHIVED\_LOG
      - Archive log 정보 표시

- Archive log 대상 지정 (\$TB\_SID.tip)

- LOG\_ARCHIVE\_DEST

# Backup 수행 방법 - Offline Backup

- 티베로 종료 후 백업

- Tibero를 정상 종료한 후 OS의 Copy 명령을 이용해 datafile, logfile, contrilfile, tip file등을 백업한다.  
MOUNT 또는 OPEN 모드에서 v\$datafile, v\$logfile 뷰를 통해서 백업할 파일 정보 조회  
ARCHIVELOG 모드에서는 archive 파일도 백업
- V\$DATAFILE에서 백업 대상 파일 조회

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

```
NAME
```

```

```

```
/home/tibero6/datafile/system001.dtf
```

```
/home/tibero6/datafile/undo001.dtf
```

```
/home/tibero6/datafile/ws_data1.dtf
```

```
/home/tibero6/datafile/ws_idx1.dtf
```

```
/home/tibero6/datafile/ts_test01.dtf
```

# Backup 수행 방법 - Offline Backup

- 티베로 종료 후 백업(계속)

- V\$LOGFILE에서 백업 대상 logfile 조회

```
SQL> select group#, member from v$logfile;
```

| GROUP# | MEMBER                            |
|--------|-----------------------------------|
| 1      | /home/tibero6/datafile/log011.log |
| 2      | /home/tibero6/datafile/log021.log |
| 3      | /home/tibero6/datafile/log031.log |

3 rows selected.

- V\$CONTROLFILE에서 백업 대상 controlfile 조회

```
SQL> select name from v$controlfile;
```

| NAME                                 |
|--------------------------------------|
| /home/tibero6/datafile/control01.ctl |

# Backup 수행 방법 - Offline Backup

- 티베로 종료 후 백업(계속)

- 티베로를 정상 종료

```
$ tbdown
```

```
Tibero instance terminated (NORMAL mode)
```

반드시 정상 종료를 해야 한다.

- File copy

앞에서 조회한 모든 파일을 copy한다.

```
$ cp /home/tibero6/datafile/*dtf /backup/
```

```
$ cp /home/tibero6/datafile/*log /backup/
```

```
$ cp /home/tibero6/datafile/*ctl /backup/
```

# Backup 수행 방법 - Online Backup

- 티베로 운영 중 백업 (온라인 백업)

- ALTER TABLESPACE 명령으로 테이블스페이스의 datafile 백업한다.  
ARCHIVELOG 모드에서만 사용 가능하다.

Tibero 데이터베이스에 온라인 백업 시작을 알림

```
SQL> ALTER TABLESPACE SYSTEM BEGIN BACKUP;
```

OS 명령으로 해당 테이블스페이스의 데이터파일 복사

```
SQL> !cp /data01/tibero/system001.tbf /backup/tibero/system001.tbf_backup
```

Tibero 데이터베이스에 온라인 백업 종료를 알림

```
SQL> ALTER TABLESPACE SYSTEM END BACKUP;
```

- 주의 사항

온라인 백업 중에는 데이터베이스의 변경 사항에 대한 log의 양이 늘어나기 때문에 가능하면 신속하게 작업을 종료해야 함

# Backup 수행 방법 - Online Backup

- 티베로 Begin backup 이후 발생 내용.

1) begin backup 실행시 redo log 에 쌓이는 로그 차이점.

→ begin backup 이 일어난 후 block 변경시 해당 block 전체를 redo log 로 남기는 image logging 이 발생함에 따라 log 양이 평소에 비해 증가하게 됨.

2) end backup 을 실행하지 않는 경우 문제점.

→ 1) 에서 언급한 바와 같이 log 양이 증가함.

→ 티베로 정상 종료가 불가하며, 만일 강제 종료할 경우 다음 부팅시 media recovery 를 해줘야 함.

3) end backup 실행시 tiberio 내부적으로 처리되는 내용.

→ controlfile 과 datafile 에서 해당 tablespace 가 backup mode 라는 플래그를 off 하며, check point 정보를 최신으로 업데이트 합니다.

4) begin backup 실행시 datafile 의 변경사항

(1) datafile 의 header 에 backup mode 라는 플래그가 on 됨.

(2) begin backup 과 end backup 사이에 발생하는 변경사항이 data file 과 redo log file 에 기록되지만, header 에 setting 되는 checkpoint 값, TSN 값은 고정된다.

# Backup 수행 방법 - Online Backup

## ● 동적 뷰

-V\$BACKUP : 현재 begin backup으로 인해 backup mode인 상태를 확인한다.

```
SQL> alter tablespace ws_data begin backup;
Tablespace altered.
```

```
SQL> select * from v$backup;
```

| FILE# | STATUS     | CHANGE# | TIME       |
|-------|------------|---------|------------|
| 0     | NOT ACTIVE | 0       |            |
| 1     | NOT ACTIVE | 0       |            |
| 2     | ACTIVE     | 270842  | 2007/10/19 |
| 3     | NOT ACTIVE | 0       |            |
| 4     | NOT ACTIVE | 0       |            |
| 5     | NOT ACTIVE | 0       |            |
| 6     | NOT ACTIVE | 0       |            |
| 7     | NOT ACTIVE | 0       |            |

8 rows selected.

# Backup 수행 방법 - Logical Backup

## ● tbExport 유틸리티

- tbExport 유틸리티를 이용해서 스키마 전체 혹은 일부를 추출해서 백업한다.
- 문제점
  - 서로 다른 시점의 데이터
  - Export 받은 시점에서의 복구만 가능하다.

## ● Export 모드

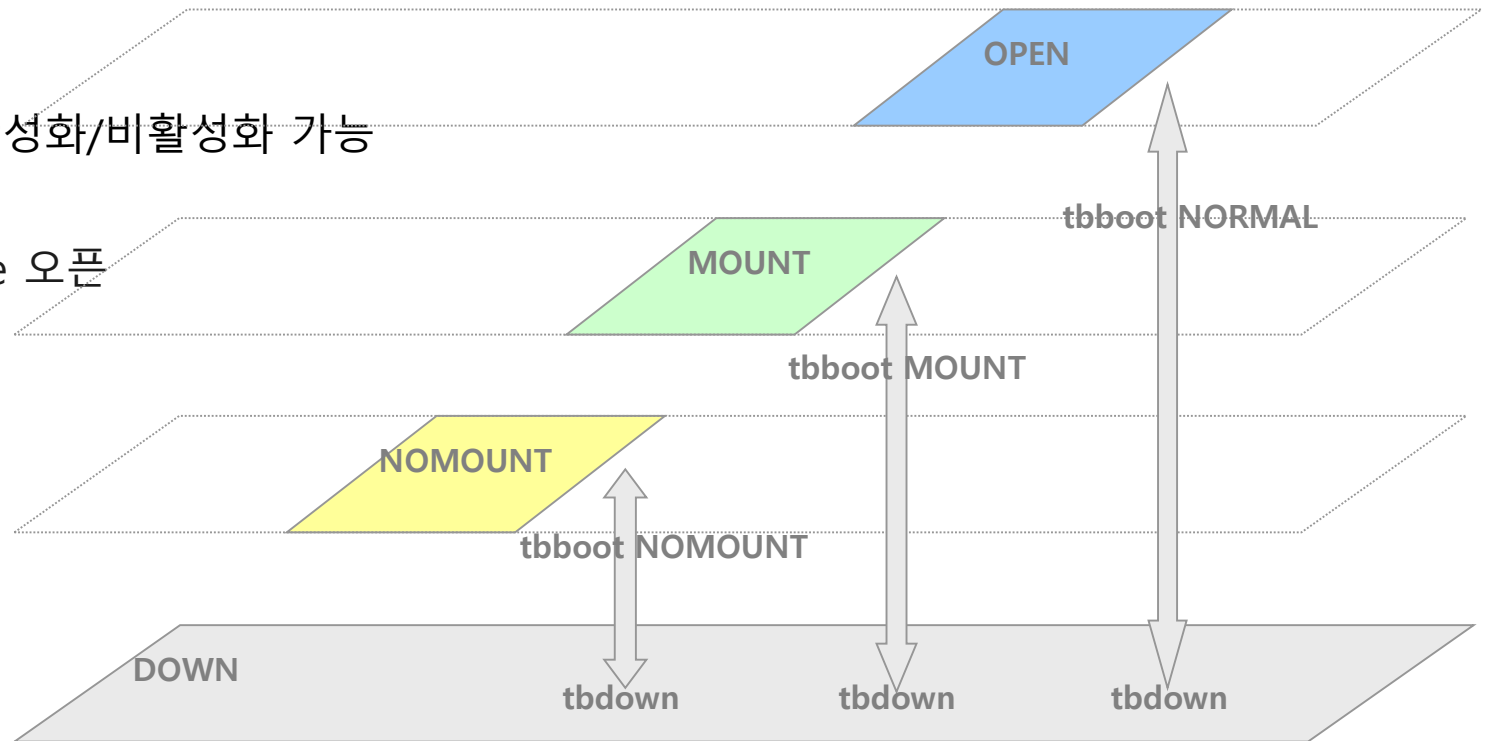
- 전체 데이터베이스 모드
  - DBA 유저(SYS)로만 가능
  - SYS를 제외한 모든 유저의 데이터를 export
  - FULL=Y
- 사용자 모드
  - 사용자가 소유하고 있는 테이블만 추출
  - USER=SCOTT
- 테이블 모드
  - 특정 테이블만 지정해서 추출
  - TABLE=SCOTT.EMP, SCOTT.DEPT



# Recovery - 인스턴스 기동 단계 별 복구 대상

## ● 부트 과정별 복구

- NOMOUNT : 인스턴스 시작  
Database 생성가능  
Controlfile 재생성 가능
- MOUNT : Controlfile 오픈  
Datafile 이름 변경가능  
Online Redo logfile Archive 옵션 활성화/비활성화 가능  
전체 데이터베이스 복구 작업 가능
- OPEN : Controlfile에 정의한 모든 file 오픈



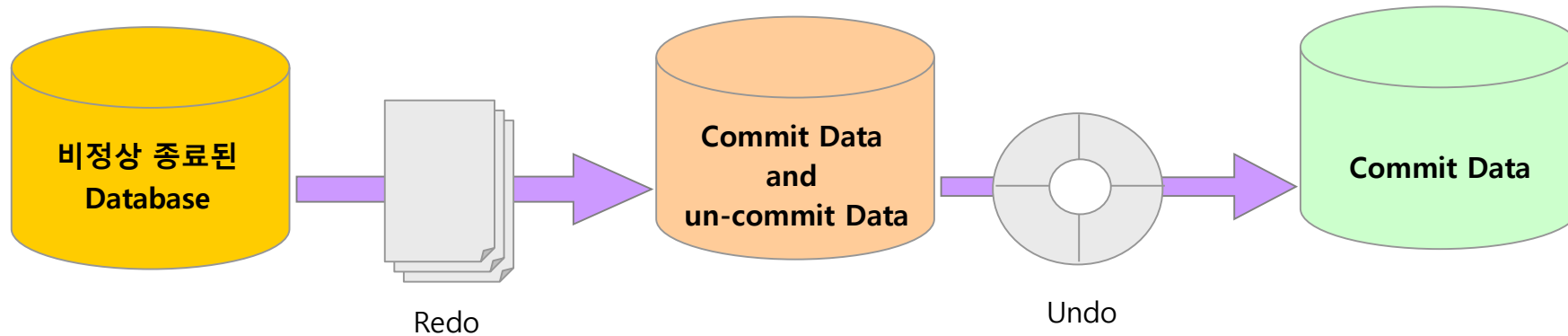
# Recovery - Crash Recovery 개요

## ● 개요

- 시스템이상, shutdown abort 등 비정상적인 데이터베이스 종료 후 Tibero가 기동하는 과정에서 자동으로 수행
- Online redo log file, online data file, current control file만을 사용
- Undo 테이블스페이스를 이용하여 commit 되지 않은 데이터에 대해 복구 작업

## ● 특징

- Database에 접근 가능한 Instance에 Failure가 발생한 경우.
- Database가 기동될 때 자동 수행됨.
- redo의 roll forward and roll back



# Recovery - Crash Recovery 원리

## ● Roll Forward

- Redo log에 존재하는 redo record를 data file에 reapply하는 작업
- Redo log 적용 순서는 Checkpoint 시점의 Lowest TSN부터 recover를 시작하여 큰 값의 TSN 순으로 진행함
- Redo Record에는 rollback(undo) segment를 변경하는 내용도 포함되어 있으므로 실제 rollback(undo) data도 발생한다.
- Roll forward 종료 시 committed change와 uncommitted change가 모두 data block에 반영되어 있게 된다. -> Roll Back Phase

## ● Roll Back

- rollback segments에 있는 undo information을 이용하여 uncommitted changes에 대한 undo 작업 수행(Consistency)

# Recovery - Media Recovery 개요

## ● Media Recovery란?

- 티베로를 구성하는 파일에 물리적인 손상이 발생하였거나 정상 동작을 할 수 없는 경우 복구하는 과정

## ● 특징

- DBA에 의한 명령에 의해 수동으로 수행 (ALTER DATABASE RECOVER ...)
- Backup받은 datafile을 이용해서 복구
- redo logfile 이나 archive logfile로부터 redo record를 적용
- MOUNT 모드에서만 가능하다.

## ● 참고하는 동적 뷰

- V\$RECOVER\_FILE
- V\$RECOVERY\_FILE\_STATUS
- V\$LOGFILE, V\$CONTROLFILE, V\$LOG

## ● 종류

- Complete Recovery : Archive logfile 과 Online logfile 을 모두 사용해서 가장 최근 로그까지 모두 반영
- Incomplete Recovery : logfile 일부만 적용 하거나 특정시점으로 복구 가능. RESETLOGS 적용 필요

# Recovery - Media Recovery 수행시 불완적 복구

## ● Incomplete Recovery

- point-in-time recovery
- redo record 의 일부 만 적용

## ● 필요한 상황

- online redo logs are damaged
- data loss by user error
- some archived log is missing

## ● Resetlogs

- Incomplete recovery를 하게 되면 반드시 resetlogs로 데이터베이스를 기동해야 한다.
- resetlogs 이전 datafile, logfile 과 resetlogs 이 후의 파일은 서로 호환되지 않는다  
resetlogs 이전에 백업 파일이나 logfile들을 이용하여 resetlogs 이후로 복구할 수 없다.  
또한 resetlogs 이후의 file들을 가지고 resetlogs 이전 상태로 incomplete 복구도 불가능하다
- resetlogs 시작한 경우, 반드시 새로운 백업을 받기를 권장한다
- Resetlogs로 데이터베이스 기동하기

```
$ tbboot -t RESETLOGS
```

A blue abstract graphic with a textured, brush-stroke-like appearance, located in the top-left corner of the slide.

2.

## 백업 및 복구 예제 1

# Controlfile 백업 및 복구

## ● Controlfile 백업

- 데이터베이스의 구조에 변화가 일어난 경우 controlfile을 백업해야 한다.
- Offline Full Backup을 하거나 trace를 통한 controlfile 생성 구문을 백업한다.
- MOUNT,OPEN 모드에서 controlfile 생성 구문 백업 가능
- 지정한 디렉토리에 controlfile 백업 생성
  - 디렉토리 경로를 지정하지 않을 경우 DB\_CREATE\_FILE\_DEST 파라미터로 설정한 디렉토리에 백업 파일 생성

# Controlfile 백업 및 복구 - Controlfile 백업 ( offline backup )

- Controlfile 확인 ( V\$CONTROLFILE )

```
SQL> select name from v$controlfile;
```

```
NAME
```

```

/home/tibero6/tibero6/database/c1.ctl
```

```
1 row selected.
```

- Tibero 종료 후 backup 대상 위치로 copy

```
$ tbdwn
```

```
Tibero instance terminated (NORMAL mode).
```

```
$ cp /home/tibero6/tibero6/database/c1.ctl /home/tibero6/tbdata/bak/c1.ctl
```



# Controlfile 백업 및 복구 - CONTROLFILE의 이중화

## ● Controlfile의 이중화

- Tibero를 종료한 후 controlfile을 copy

```
$ tbdwn
```

```
Tibero instance terminated (NORMAL mode).
```

```
$cp /tibero/tibero6/database/c1.ctl /tibero/tibero6/database/c1.ctl
```

- \$TB\_SID.tip 파일의 CONTROL\_FILES 항목에 추가

```
CONTROL_FILES="/tibero/tibero6/database/c1.ctl", "/tibero/tibero6/database/c2.ctl"
```

# Controlfile 백업 및 복구 - Controlfile 백업 (Using trace)

- **Controlfile 백업(Online)**

- Online 백업 시, Alter ~ trace 구문을 사용하여 Controlfile 생성용 SQL문장을 백업하고, 복구시 이를 사용한다.
- ALTER DATABASE BACKUP CONTROLFILE

```
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE
AS [백업할 파일 경로 및 이름] REUSE NORESETLOGS;
```

- 실행 예)

```
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE AS
'/home/tibero6/tbdata/backup/create_controlfile.sql' REUSE NORESETLOGS;
```

# Controlfile 백업 및 복구 - Controlfile 백업 (Using trace)

## ● 백업된 컨트롤 파일의 생성문 확인 예)

```
CREATE CONTROLFILE REUSE DATABASE "tibero"
LOGFILE
GROUP 0 (
 '/home/tibero6/tbdata/log01.log',
 '/home/tibero6/tbdata/log02.log'
) SIZE 10M,
GROUP 1 (
 '/home/tibero6/tbdata/log11.log',
 '/home/tibero6/tbdata/log12.log'
) SIZE 10M,
GROUP 2 (
 '/home/tibero6/tbdata/log21.log',
 '/home/tibero6/tbdata/log22.log'
) SIZE 10M,
GROUP 3 (
 '/home/tibero6/tbdata/redo31.log',
 '/home/tibero6/tbdata/redo32.log'
) SIZE 512K
NORESETLOGS
DATAFILE
 '/home/tibero6/tbdata/system001.dtf',
 '/home/tibero6/tbdata/undo001.dtf',
 '/home/tibero6/tbdata/usr001.dtf',
 '/home/tibero6/tbdata/my_file001.dtf',
 '/home/tibero6/tbdata/my_file002.dtf'
NOARCHIVELOG
MAXLOGFILES 100
MAXLOGMEMBERS 8
MAXDATAFILES 256
CHARACTER SET MSWIN949
NATIONAL CHARACTER SET UTF16
;
```

```
--ALTER DATABASE MOUNT
```

```
---- Recovery is required in MOUNT mode.
```

```
--ALTER DATABASE RECOVER AUTOMATIC;
```

```
--ALTER DATABASE OPEN ;
```

```
---- Adding Tempfiles is required in OPEN mode.
```

```
-- ALTER TABLESPACE TEMP ADD TEMPFILE '/home/tibero6/tbdata/temp001.dtf'
```

```
-- SIZE 512M REUSE AUTOEXTEND ON NEXT 8M MAXSIZE 3G;
```

# Controlfile 백업 및 복구 - Controlfile 복구 순서

- controlfile 재생성을 통한 복구 과정 요약

- 1) NOMOUNT 모드로 기동

```
$ tbboot nomount
```

- 2) tbsql 접속 후 Control 파일 백업 받은 SQL 스크립트 실행

```
SQL> @/backup/create_controlfile.sql
```

- 3) MOUNT 모드로 재기동

```
$ tbdwn
$ tbboot mount
```

- 4) Media 복구

```
SQL> ALTER DATABASE RECOVER AUTOMATIC DATABASE;
```

- 5) NORMAL 모드로 재기동

```
$ tbdwn
$ tbboot
```

- 6) temp datafile 추가

```
SQL> alter tablespace temp add tempfile '/disk1/temp01.dtf' size 10m reuse;
```

- 7) temp datafile 확인

```
SQL> select file_id, tablespace_name from dba_temp_files;
 FILE_ID TABLESPACE_NAME
 ----- -
 0 TEMP
```

# Controlfile 백업 및 복구 - Controlfile 복구 과정 진행

- **controlfile 재생성을 통한 복구 과정 상세**

- 1) NOMOUNT 모드로 기동

: controlfile 생성작업은 nomount 모드에서 진행합니다.

```
$ tbboot nomount
```

```
listener port = 8629
```

```
change core dump dir to /home/tibero6/tibero6/bin/prof
```

```
Tibero RDBMS 6
```

```
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.
```

```
Tibero instance started suspended at NOMOUNT mode.
```

# Controlfile 백업 및 복구 - Controlfile 복구 과정 진행

- controlfile 재생성을 통한 복구 과정 상세

2) tbsql 접속 후 Control 파일 백업 받은 SQL 스크립트 실행

```
$ tbsql sys/tibero
```

```
tbSQL 5
```

```
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.
```

```
Connected to Tibero.
```

```
SQL> @cre_controlfile.sql
```

```
Control File created.
```

```
File finished.
```

# Controlfile 백업 및 복구 - Controlfile 복구 과정 진행

- controlfile 재생성을 통한 복구 과정 상세

- 3) MOUNT 모드로 재기동

```
$ tbdwn
```

```
Tibero instance terminated (NORMAL mode).
```

```
$ tbboot mount
```

```
listener port = 8629
```

```
change core dump dir to /home/tibero6/tibero6/bin/prof
```

```
Tibero RDBMS 6
```

```
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.
```

```
Tibero instance started suspended at MOUNT mode.
```

# Controlfile 백업 및 복구 - Controlfile 복구 과정 진행

- controlfile 재생성을 통한 복구 과정 상세

- 4) Media Recovery

```
$ tbsql sys/tibero
```

```
tbSQL 6
```

```
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.
```

```
Connected to Tibero.
```

```
SQL>
```

```
SQL> alter database recover automatic database;
```

```
Database altered.
```



# Controlfile 백업 및 복구 - Controlfile 복구 과정 진행

- controlfile 재생성을 통한 복구 과정 상세

- 5) Nomal mode 로 Tibero 재기동

```
$ tbdwn
Tibero instance terminated (NORMAL mode).

$ tbboot
listener port = 8629
change core dump dir to /home/tibero6/tibero6/bin/prof
Tibero RDBMS 6
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.
Tibero instance started up (NORMAL mode).
```

- 6) temp datafile 추가

```
SQL> select file_id, tablespace_name from dba_temp_files;
 FILE_ID TABLESPACE_NAME

0 selected.
```

```
SQL> alter tablespace temp add tempfile '/disk1/temp01.dtf' size 10m reuse;
```

- 7) temp datafile 확인

```
SQL> select file_id, tablespace_name from dba_temp_files;
 FILE_ID TABLESPACE_NAME

 0 TEMP
```

# Controlfile 백업 및 복구 - Controlfile 복구 (이중화)

- 이중화한 controlfile 중 일부의 손상

- 이중화한 controlfile 중 일부가 손상되어도, 하나의 controlfile 을 통해 서비스 가능함.
- 일부 controlfile 손상시  
하나의 controlfile 파일을 손상시킴(이름을 변경).

```
c1ctl
c2ctl.CRACK
```

- 하나의 controlfile을 통해 정상 동작하지만, 다음과 같이 tip 파일설정과 실제파일을 일치시키도록 함.  
\$TB\_SID.tip 파일의 CONTROL\_FILES 항목에서 손상된 controlfile의 목록을 삭제

```
#CONTROL_FILES="/home/tibero6/tbdata/c1ctl","/home/tibero6/tbdata/c2ctl"
CONTROL_FILES="/home/tibero6/tbdata/c1ctl"
```

- 손상되지 않은 controlfile에서 copy

다중화 구성한 controlfile 중에 손상되지 않은 controlfile 파일이 있다면, 이 파일을 OS 명령으로 복사해서 복구

```
$cp /home/tibero6/tbdata/c1ctl /home/tibero6/tbdata/c2ctl
```

# 오프라인 백업 및 복구

- 오프라인 백업 데이터를 이용한 복구

1. 백업

- Controlfile

\$TB\_SID.tip 파일의 CONTROL\_FILES 파라미터에서 정의되었거나 v\$controlfile에서 controlfile 목록 확인

- Datafile

```
SQL> SELECT NAME FROM V$CONTROLFILE;
```

MOUNT 또는 OPEN 모드에서 v\$datafile 뷰를 통해 검색한 데이터 파일 백업

- Logfile

```
SQL> SELECT NAME FROM V$DATAFILE;
```

MOUNT 또는 OPEN 모드에서 v\$logfile 에서 검색한 로그 파일을 각 그룹별로 하나의 멤버 파일만 백업

```
SQL> SELECT GROUP#, MEMBER FROM V$LOGFILE;
```

- 파라미터 파일

\$TB\_SID.tip 파일 백업

# 오프라인 백업 및 복구

- 오프라인 백업 데이터를 이용한 복구

- 2. 장애(일부 데이터 파일 삭제)

- Tibero 종료 후 동일 시점에 백업 받은 Controlfile, Datafile, logfile 을 OS 명령으로 복사해서 복구
    - 복구 예) 데이터파일 중 일부 삭제된 장애

```
$ rm usr001.dtf

$ tbboot
listener port = 8629
change core dump dir to /home/tibero6/tibero6/bin/prof

* Critical Warning : Raise svmode failed. The reason is
* TBR-1024 : Database needs media recovery: open failed(/tibero/tibero6/database/usr001.dtf).
* Current server mode is MOUNT.

Tibero RDBMS 6

Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.

Tibero instance started suspended at MOUNT mode.
```

# 오프라인 백업 및 복구

- 오프라인 백업 데이터를 이용한 복구

- 3. 데이터베이스 리스토어( 백업 데이터를 이용 )

- 기존 데이터베이스 파일 전체 삭제 ( 일관성을 위해 전체 파일을 백업 파일로 교체 )
    - Tibero를 종료 한 후 backup 받은 datafile을 원래의 위치로 copy

```
$ tbdown
```

```
Tibero instance terminated (NORMAL mode).
```

```
$ rm /tiberio/tibero6/backup/*.ctl
```

```
$ rm /tiberio/tibero6/backup/*.log
```

```
$ rm /tiberio/tibero6/backup/*.dtf
```

```
$ cp /tiberio/tibero6/backup/*.ctl /tiberio/tibero6/database
```

```
$ cp /tiberio/tibero6/backup/*.log /tiberio/tibero6/database
```

```
$ cp /tiberio/tibero6/backup/*.dtf /tiberio/tibero6/database
```

# 오프라인 백업 및 복구

- 오프라인 백업 데이터를 이용한 복구

## 4. Tibero 기동

```
$ tbboot
```

```
listener port = 8629
```

```
change core dump dir to /home/tibero6/tibero6/bin/prof
```

```
Tibero RDBMS 6
```

```
Copyright (c) 2008, 2009, 2011 Tiberio Corporation. All rights reserved.
```

```
Tibero instance started suspended at NORMAL mode.
```

# Database 파일 위치 변경 - controlfile

- Tibero 구성파일 위치 변경 - controlfile

- Controlfile 위치 변경

Tibero 종료

```
$ tbdwn
```

```
Tibero instance terminated (NORMAL mode).
```

\$TB\_SID.tip 파일에서 CONTROL\_FILES의 파일 경로 또는 이름변경

```
#CONTROL_FILES="/tibero/tibero6/database/c1.ctl"
```

```
CONTROL_FILES="/tibero/tibero6/database/control01.ctl"
```

변경하고자 하는 파일에 대해 OS 명령으로 파일 경로 또는 이름변경후, TIBERO 기동

```
$ mv c1.ctl control01.ctl
```

```
$ tbboot
```

```
listener port = 8629
```

```
change core dump dir to /home/tibero6/tibero6/bin/prof
```

```
Tibero RDBMS 6
```

```
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.
```

```
Tibero instance started suspended at NORMAL mode.
```

# Database 파일 위치 변경 - datafile, logfile

- Tibero 구성파일 위치 변경 – datafile, logfile

- Datafile, Logfile rename

- 1) 변경 전 파일 정보 조회 ( V\$DATAFILE, V\$LOGFILE 에서 변경할 file 정보 확인 )

```
SQL> SELECT MEMBER FROM V$LOGFILE;
```

```
SQL> SELECT NAME FROM V$DATAFILE;
```

- 2) TIBERO 정상 종료 ( tbdownd immediate 수행 )

```
$ tbdownd immediate
```

- 3) OS 명령으로 파일의 경로 또는 이름 변경

```
$ mv /tiberio/tbdata/system001.dtf /tiberio/tbdata2/system001.dtf
```

- 4) TIBERO 를 MOUNT 모드로 기동

```
$ tbboot mount
```

- 5) Controlfile 의 정보 변경 ( RENAME FILE '기존 경로이름' TO '변경된 경로이름' )

```
SQL> ALTER DATABASE RENAME FILE '/tiberio/tbdata/system001.dtf' TO '/tiberio/tbdata2/system001.dtf'
```

- 6) 변경 후 파일 정보 조회 ( V\$DATAFILE, V\$LOGFILE 에서 변경 이후 file 정보 확인 )

```
SQL> SELECT MEMBER FROM V$LOGFILE;
```

```
SQL> SELECT NAME FROM V$DATAFILE;
```



A blue abstract graphic with a textured, brush-stroke-like appearance, located in the top-left corner of the slide.

3.

## 백업 및 복구 예제 2

# ARCHIVELOG MODE 변경

## < 아카이브 로그 모드 변경 작업 순서 >

1. Tibero 종료
2. 파라미터 파일에 LOG\_ARCHIVE\_DEST 설정
3. Tibero mount 모드 기동
4. 아카이브로그 모드 변경 쿼리 수행
5. Tibero normal 모드 기동
6. 아카이브 모드 확인
7. 아카이브로그 생성하여 파일 확인

# ARCHIVELOG MODE 변경

## 1. Tibero 종료

```
$ tbdown
```

Tibero instance terminated (NORMAL mode).

## 2. 파라미터 파일에 LOG\_ARCHIVE\_DEST 설정

```
$ vi $TB_HOME/config/$TB_SID.tip
```

```
....
```

```
LOG_ARCHIVE_DEST="/tibero/tbdata/tibero/arch"
```

## 3. Tibero mount 모드 기동

```
$ tbboot mount
```

Tibero instance started up (MOUNT mode).

# ARCHIVELOG MODE 변경

## 4. 아카이브로그 모드 변경 쿼리 수행

```
SQL> alter database archivelog;
```

## 5. Tibero normal 모드 기동

```
$ tbdwn
```

```
$ tbboot
```

```
Tibero instance started up (NORMAL mode).
```

## 6. 아카이브 모드 확인

- V\$DATABASE 조회

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

```
LOG_MODE
```

```

```

```
ARCHIVELOG
```

```
SQL> ARCHIVE LOG LIST
```

| NAME | VALUE |
|------|-------|
|------|-------|

|                              |                            |
|------------------------------|----------------------------|
| Database log mode            | Archive Mode               |
| Archive destination          | /tibero/tbdata/tiber/arch/ |
| Oldest online log sequence   | 1                          |
| Next log sequence to archive | 2                          |
| Current log sequence         | 2                          |

# ARCHIVELOG MODE 변경

## 6. 로그 스위치 실행

```
SQL> ALTER SYSTEM SWITCH LOGFILE;
```

## 7. 아카이브 모드 확인

- ARCHIVE LOG LIST 명령으로 조회

```
SQL> ARCHIVE LOG LIST
```

| NAME                         | VALUE                       |
|------------------------------|-----------------------------|
| Database log mode            | Archive Mode                |
| Archive destination          | /tibero/tbdata/tibero/arch/ |
| Oldest online log sequence   | 1                           |
| Next log sequence to archive | 2                           |
| Current log sequence         | 2                           |

## 6. 아카이브로그 생성하여 파일 확인

- V\$ARCHIVE\_DEST\_FILES 조회

```
SQL> COL NAME FOR A60
```

```
SQL> SELECT SEQUENCE#, NAME FROM V$ARCHIVE_DEST_FILES;
```

| SEQUENCE# | NAME                                        |
|-----------|---------------------------------------------|
| 1         | /tibero/tbdata/tibero/arch/log-t0-r0-s1.arc |

- V\$ARCHIVED\_LOG 조회

```
SQL> SELECT SEQUENCE#, NAME FROM V$ARCHIVED_LOG;
```

| SEQUENCE# | NAME                                        |
|-----------|---------------------------------------------|
| 1         | /tibero/tbdata/tibero/arch/log-t0-r0-s1.arc |

# Redo Logfile 장애시 복구

- logfile 장애 복구

- CASE 1 : 리두로그가 다중화 되어 있는 환경에서 멤버 중 일부가 장애
- CASE 2 : 특정 리두로그 그룹의 모든 멤버 파일 장애 ( CURRENT 가 아닌 경우 )
- CASE 3 - logfile 장애 복구 ( current logfile 일때 )

# Redo Logfile 장애시 복구

- CASE 1 : 리두로그가 다중화 되어 있는 환경에서 멤버 중 일부가 장애
    - 장애 발생 : Current logfile이 아닌 Logfile 을 삭제 한다.
    - 복구 방법 : 장애가 발생한 logfile이 속한 group의 다른 멤버 logfile을 복사하여 복원
1. 티베로 종료 ( tbdwn immediate )
  2. logfile 생성 ( 다른 멤버의 로그파일 -> 삭제된 멤버 파일 copy )
  3. 티베로 기동 ( tbboot )

# Redo Logfile 장애시 복구

- CASE 2 : 특정 리두로그 그룹의 모든 멤버 파일 장애 ( CURRENT 가 아닌 경우 )

- 1) 장애 발생 : CURRENT 가 아닌 그룹의 멤버 파일 모두 삭제후, 티베로 종료 ( tbdownd immediate )
- 2) 티베로 시작 : tbboot 수행시, 일부 logfile 이 없으므로 MOUNT 모드로 기동 하게 됨
- 3) 문제가 발생한 로그파일 Group 삭제 (V\$LOGFILE 참조)
- 4) 새로운 로그파일 Group 추가
- 5) NORMAL 모드로 재기동



# Redo Logfile 장애시 복구

- CASE 2 - logfile 장애 복구 ( current logfile 아닌 경우 )

- 1) 장애 발생 : CURRENT 가 아닌 그룹의 멤버 파일 모두 삭제후, 티베로 종료 ( tbdowm immediate )
- 2) 티베로 시작 : tbboot 수행시, 일부 logfile 이 없으므로 MOUNT 모드로 기동 하게 됨

```
$ tbboot
listener port = 8629
change core dump dir to /home/tibero6/tibero6/instance/tibero
```

```

* Critical Warning : Raise svmode failed. The reason is
* TBR-1003 : Cannot open file: /home/tibero6/tbdata/redo11.redo.
* Current server mode is MOUNT.

```

Tibero RDBMS 6

Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.

Tibero instance started suspended at MOUNT mode.

# Redo Logfile 장애시 복구

- CASE 2 - logfile 장애 복구 ( current logfile 아닌 경우 )
  - 3) 문제가 발생한 로그파일 Group 삭제 (V\$LOGFILE 참조)

```
SQL> select group#, member from v$logfile;
```

```
GROUP# MEMBER
```

```

```

```
0 /home/tibero6/tbdata/redo01.redo
```

```
1 /home/tibero6/tbdata/redo11.redo
```

```
2 /home/tibero6/tbdata/redo21.redo
```

```
3 rows selected.
```

```
SQL> alter database drop logfile group 1;
```

```
Database altered.
```

```
SQL>
```

# Redo Logfile 장애시 복구

- CASE 2 - logfile 장애 복구 ( current logfile 아닌 경우 )

- 4) 새로운 logfile group 추가

```
SQL> ALTER DATABASE ADD LOGFILE GROUP 1 ('/home/tibero6/tbdata/redo11.redo') size 40M;
```

```
Database altered.
```

```
SQL>
```

# Redo Logfile 장애시 복구

- CASE 2 - logfile 장애 복구 ( current logfile 아닌 경우 )

5) Tibero를 재기동한다.

```
$ tbdwn
```

```
Tibero instance terminated (NORMAL mode).
```

```
$ tbboot
```

```
listener port = 8629
```

```
change core dump dir to /home/tibero6/tibero6/instance/tibero
```

```
Tibero RDBMS 6
```

```
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.
```

```
Tibero instance started suspended at NORMAL mode.
```

```
$
```

# Redo Logfile 장애시 복구

- CASE 3 - logfile 장애 복구 ( current logfile 일때 )

- 1) 장애 발생 : CURRENT 그룹의 멤버 파일 모두 삭제후, 티베로 종료( **tbdowndown immediate** )
- 2) 티베로 시작 : tbboot 수행시, 일부 logfile 이 없으므로 MOUNT 모드로 기동 하게 됨
- 3) 리두 로그 조회
- 4) 복구 명령 수행 ( ALTER DATABASE RECOVER ~ )
- 5) RESETLOGS 모드로 재기동

\*\* 1) 번에서 비정상 종료 수행된 경우는, 4) 번의 복구 작업시 과거에 온라인 백업받은 데이터 파일이 필요함.

# Redo Logfile 장애시 복구

- CASE 3 - logfile 장애 복구 ( current logfile 일때 )

1) 장애 발생 : CURRENT 그룹의 멤버 파일 모두 삭제후, 티베로 종료( tbdownd immediate )

```
SQL> select group#, status from v$log;
```

```
GROUP# STATUS
```

```

0 INACTIVE
```

```
1 INACTIVE
```

```
2 CURRENT
```

```
SQL> SELECT MEMBER FROM V$LOGFILE WHERE GROUP#=2;
```

```
MEMBER
```

```

/tibero/tbdata/tibero/log005.log
```

```
/tibero/tbdata/tibero/log006.log
```

```
SQL> !rm /tibero/tbdata/tibero/log005.log /tibero/tbdata/tibero/log006.log
```

```
SQL> tbdownd immediate
```

```
DB instance was terminated.
```

# Redo Logfile 장애시 복구

- CASE 3 - logfile 장애 복구 ( current logfile 일때 )

2) 티베로 시작 : tboot 수행시, 일부 logfile 이 없으므로 MOUNT 모드로 기동 하게 됨

```
[tibero@T1:tibero]$ tboot
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
```

```
Listener port = 8629
```

```

```

```
* Critical Warning : Raise svmode failed. The reason is
```

```
* TBR-1042 : Unable to read log member file in group 2, member -1 (), block 0.
```

```
* Current server mode is MOUNT.
```

```

```

Tibero 6

TmaxData Corporation Copyright (c) 2008-. All rights reserved.

Tibero instance started suspended at **MOUNT** mode.

# Redo Logfile 장애시 복구

- CASE 3 - logfile 장애 복구 ( current logfile 일때 )

- 3) 리두 로그 조회

```
SQL> select group#, status from v$log;
```

```
GROUP# STATUS
```

```

0 INACTIVE
```

```
1 INACTIVE
```

```
2 CURRENT
```

```
SQL> SELECT MEMBER FROM V$LOGFILE WHERE GROUP#=2;
```

```
MEMBER
```

```

/tibero/tbdata/tibero/log005.log
```

```
/tibero/tbdata/tibero/log006.log
```

```
SQL> !ls -l /tibero/tbdata/tibero/log00*.log
```

```
-rwxr-x--- 1 tibero dba 52428800 Apr 2 16:01 /tibero/tbdata/tibero/log001.log
```

```
-rwxr-x--- 1 tibero dba 52428800 Apr 2 16:01 /tibero/tbdata/tibero/log002.log
```

```
-rwxr-x--- 1 tibero dba 52428800 Apr 2 16:01 /tibero/tbdata/tibero/log003.log
```

```
-rwxr-x--- 1 tibero dba 52428800 Apr 2 16:01 /tibero/tbdata/tibero/log004.log
```



# Redo Logfile 장애시 복구

- CASE 3 - logfile 장애 복구 ( current logfile 일때 )

- 4) 복구 명령 수행 ( ALTER DATABASE RECOVER ~ )

```
SQL> ALTER DATABASE RECOVER DATABASE UNTIL CANCEL;
```

```
Database altered.
```

- 5) RESETLOGS 모드로 재기동

```
[tibero@T1:/tibero]$ tbdown
```

```
Tibero instance terminated (NORMAL mode).
```

```
[tibero@T1:/tibero]$ tbboot resetlogs
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
```

```
Listener port = 8629
```

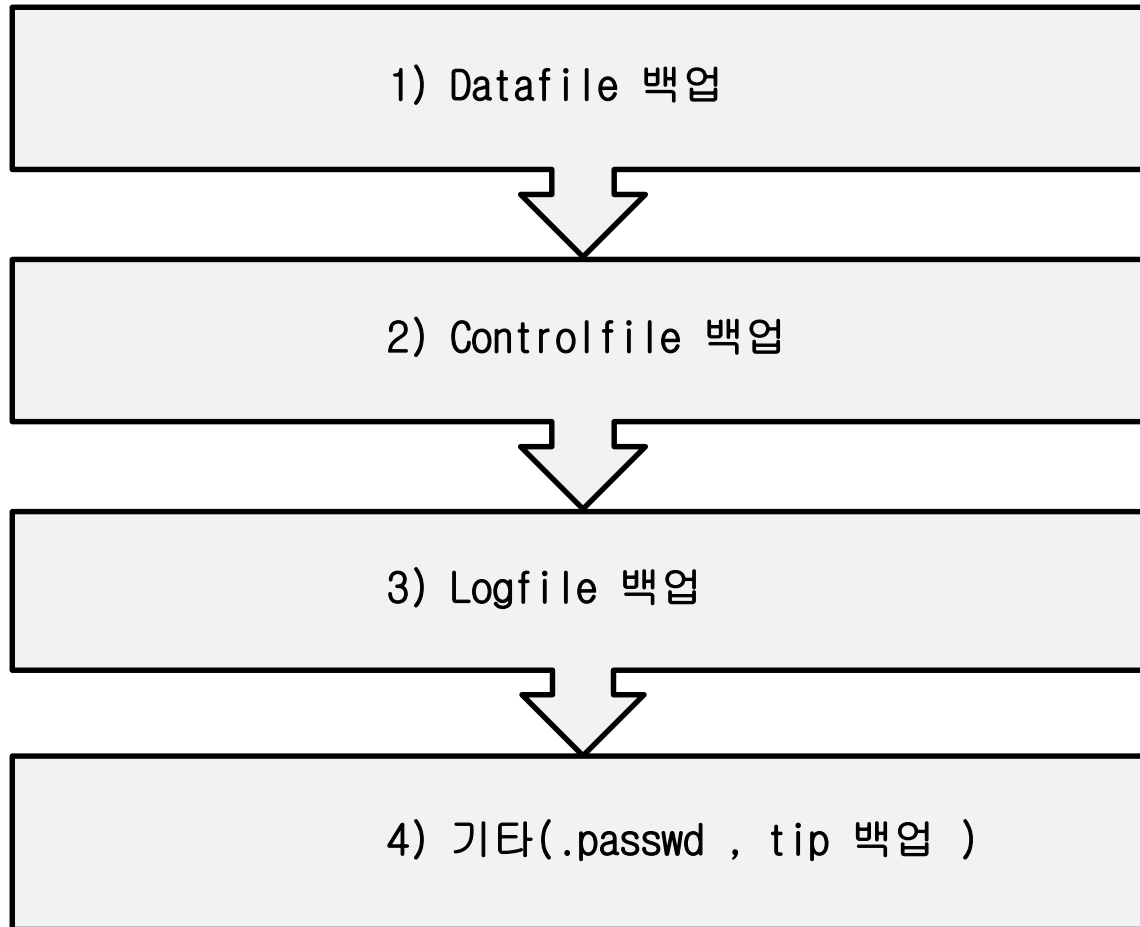
```
Tibero 6
```

```
TmaxData Corporation Copyright (c) 2008-. All rights reserved.
```

```
Tibero instance started up (NORMAL RESETLOGS mode).
```

# Online Backup

- Online backup : 인스턴스가 데이터베이스를 open 하여 사용중에 Backup 을 수행하는 것을 말함.
- 작업 순서



# Online Backup

## 1) Datafile 백업

### 1) Tablespace 조회

```
SQL> SELECT NAME FROM V$TABLESPACE;
```

### 2) Begin backup 실행 ( 테이블스페이스를 백업 모드로 변경함)

```
SQL> ALTER TABLESPACE <테이블스페이스 이름> BEGIN BACKUP;
```

참고)

TABLESPACE 들에 대해 반복 수행함  
( 2 ~ 4 과정을 반복 수행 )

### 3) Datafile 백업 ( 백업 경로로 Copy )

```
SQL> !cp /tibero/tbdata/tibero/usr001.dtf /tibero/s/backup/tbdata/tibero/usr001.dtf
```

### 4) END backup 실행 ( 테이블스페이스를 백업 모드를 종료함 )

```
SQL> ALTER TABLESPACE <테이블스페이스 이름> END BACKUP;
```

### 5) Log Switch 수행

```
SQL> ALTER SYSTEM SWITCH LOGFILE;
```

# Online Backup

## 2) Controlfile 백업

- 백업 명령 실행 ( ALTER DATABASE ~ TRACE AS '컨트롤파일 생성용 스크립트파일 경로와 이름' )

```
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE AS '/tibero/s/backup/online/crectl.sql'
REUSE NORESETLOGS;
```

## 3) Logfile 백업

- 1) Log switch 실행

```
SQL> ALTER SYSTEM SWITCH LOGFILE;
```

- 2) Archivelog 파일 백업

```
SQL> !cp /tibero/tbdata/tibero/arch/*.arc /tibero/s/backup/tbdata/tibero/arch/*.arc
```

## 4) 기타 백업

- 패스워드 파일 : DB\_CREATE\_FILE\_DEST 경로의 .passwd 파일
- 파라미터 파일 : \$TB\_HOME/config/\$TB\_SID.tip

# Online Backup - Tibero의 재기동시 복구

- Online backup 중 비정상적인 종료 등으로 tibero의 재기동하게 되면 에러 발생

1. 장애 - Tibero 기동시 MOUNT 모드까지 boot 진행됨
2. (MOUNT 모드) 백업 상태 조회
3. (MOUNT 모드) 백업 모드 종료 수행
4. 데이터베이스 오픈

# Online Backup - Tibero의 재기동시 복구

## 1. 장애 - Tibero 기동시 MOUNT 모드까지 boot 진행됨

-Online backup 중 비정상적인 종료 등으로 tibero의 재기동하게 되면 에러 발생

```
$ tbboot
listener port = 8629
change core dump dir to /home/tibero6/tibero6/instance/tibero

* Critical Warning : Raise svmode failed. The reason is
* TBR-1024 : Database needs media recovery: file restored(/home/tibero6/tbdata/system001.dtf).
* Current server mode is MOUNT.

Tibero RDBMS 6

Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.

Tibero instance started suspended at MOUNT mode.
```

# Online Backup - Tibero의 재기동시 복구

## 2. (MOUNT 모드) 백업 상태 조회

–V\$BACKUP view를 조회하여 STATUS가 'ACTIVE'인 tablespace를 조회

```
SQL> SELECT * FROM V$BACKUP;
```

| FILE# | STATUS     | CHANGE# | TIME       |
|-------|------------|---------|------------|
| ----- |            |         |            |
| 0     | ACTIVE     | 36725   | 2019/04/02 |
| 1     | NOT ACTIVE | 36741   | 2019/04/02 |
| 2     | NOT ACTIVE | 36752   | 2019/04/02 |
| 3     | NOT ACTIVE | 36761   | 2019/04/02 |

# Online Backup - Tibero의 재기동시 복구

## 3. (MOUNT 모드) 백업 모드 종료 수행

- "ALTER TABLESPACE [tablespace\_name] END BACKUP; " 구문 실행

```
SQL> ALTER TABLESPACE SYSTEM END BACKUP;
```

- 백업 상태 조회 : NOT ACTIVE 로 조회 됨을 확인

```
SQL> SELECT * FROM V$BACKUP;
```

| FILE# | STATUS     | CHANGE# | TIME       |
|-------|------------|---------|------------|
| ----- | -----      | -----   | -----      |
| 0     | NOT ACTIVE | 36983   | 2019/04/02 |
| 1     | NOT ACTIVE | 36741   | 2019/04/02 |
| 2     | NOT ACTIVE | 36752   | 2019/04/02 |
| 3     | NOT ACTIVE | 36761   | 2019/04/02 |

## 4. 데이터베이스 오픈 재기동 Tibero를 재기동한다.

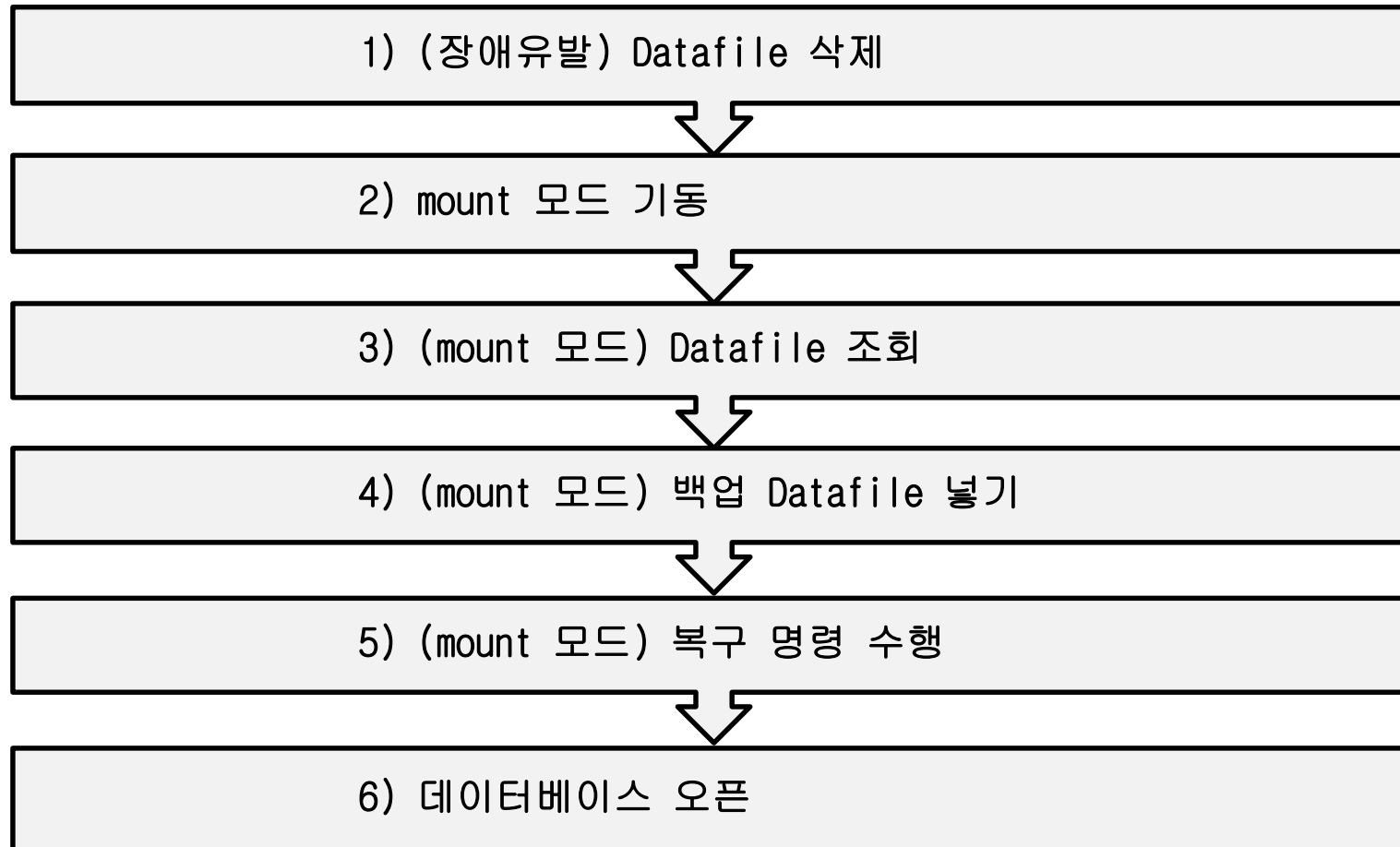
```
$ tbdwn
```

```
$ tbboot
```



# 백업 파일을 이용한 datafile 완전 복구

- 장애 내용 : 일부 Datafile 손상
- 복구 방법 : mount 모드에서 손상된 datafile 을 온라인 백업본으로 교체후, 완전 복구를 수행함
- 작업 순서



# 백업 파일을 이용한 datafile 완전 복구

## 1) (장애유발) Datafile 삭제

-Datafile을 삭제한 후 TIBERO 비정상 종료

```
SQL> SELECT NAME FROM V$DATAFILE;
```

```
NAME
```

```

/tibero/tbdata/tibero/system001.dtf
/tibero/tbdata/tibero/undo001.dtf
/tibero/tbdata/tibero/usr001.dtf
/tibero/tbdata/tibero/tpr_ts.dtf
```

```
SQL> !rm /tibero/tbdata/tibero/usr001.dtf
```

```
SQL> quit
```

```
Disconnected.
```

```
[tibero@T1:/tibero]$ tbdwn abnormal
```

```
Tibero instance terminated (ABNORMAL mode).
```

# 백업 파일을 이용한 datafile 완전 복구

## 2) mount 모드 기동

- 티베로 시작시, MOUNT 모드로 기동됨( 에러메시지 발생 )

```
[tibero@T1:/tibero]$ tbboot
Change core dump dir to /tibero/tibero6/bin/prof.
Listener port = 8629

* Critical Warning : Raise svmode failed. The reason is
* TBR-1024 : Database needs media recovery: open failed(/tibero/tbdata/tibero/usr001.dtf).
* Current server mode is MOUNT.

Tibero 6

TmaxData Corporation Copyright (c) 2008-. All rights reserved.
Tibero instance started suspended at MOUNT mode.
```

# 백업 파일을 이용한 datafile 완전 복구

## 3) (mount 모드) 데이터 파일 조회

- open faild 가 발생한 데이터 파일을 조회함

```
SQL> COL TIME FOR A10
```

```
SQL> COL ERROR FOR A30
```

```
SQL> SELECT * FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR       | CHANGE# | TIME |
|-------|--------|-------------|---------|------|
| 2     | ONLINE | open failed | 0       |      |

```
SQL> SELECT NAME FROM V$DATAFILE WHERE FILE# = 2;
```

```
NAME
```

```

/tibero/tbdata/tibero/usr001.dtf
```

```
SQL> !ls /tibero/tbdata/tibero/usr001.dtf
```

```
ls: cannot access /tibero/tbdata/tibero/usr001.dtf: No such file or directory
```

# 백업 파일을 이용한 datafile 완전 복구

## 4) (mount 모드) 백업 파일 넣기

- 장애 이전에 백업하였던 해당 데이터파일 "usr001.dtf" 을 넣음

```
SQL> !cp /tibero/s/backup/online/tbdata/tibero/usr001.dtf /tibero/tbdata/tibero
```

- V\$RECOVER\_FILE 조회

```
SQL> COL TIME FOR A10
```

```
SQL> COL ERROR FOR A30
```

```
SQL> SELECT * FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR         | CHANGE# | TIME       |
|-------|--------|---------------|---------|------------|
| 2     | ONLINE | file restored | 36296   | 2019/04/03 |

# 백업 파일을 이용한 datafile 완전 복구

## 5) (mount 모드) 복구 명령 수행

- 미디어 복구 명령 실행함.

```
SQL> ALTER DATABASE RECOVER AUTOMATIC DATABASE;
```

```
Database altered.
```

- V\$RECOVER\_FILE 조회

```
SQL> COL TIME FOR A10
```

```
SQL> COL ERROR FOR A30
```

```
SQL> SELECT * FROM V$RECOVER_FILE;
```

```
0 row selected.
```

# 백업 파일을 이용한 datafile 완전 복구

## 6) 데이터베이스 오픈

- tbdwn, tbboot 명령으로 재기동하여, 인스턴스가 NORMAL mode 로 구동되는지 확인한다.

```
[tibero@T1:/tibero]$ tbdwn
```

```
Tibero instance terminated (NORMAL mode).
```

```
[tibero@T1:/tibero]$ tbboot
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
```

```
Listener port = 8629
```

```
Tibero 6
```

```
TmaxData Corporation Copyright (c) 2008-. All rights reserved.
```

```
Tibero instance started up (NORMAL mode).
```

# 백업 파일이 없는 datafile 완전 복구

- 장애 내용 : 일부 Datafile 손상 (백업 Datafile 없음)
- 복구 방법 : mount 모드에서 손상된 datafile 을 신규 생성하여, 완전 복구를 수행함

- 작업 순서

| 1) (장애유발) Datafile 삭제     | SQL> <b>!rm /tiberio/tbdata/tiberio/usr001.dtf</b><br>SQL> <b>!tbdwn abnormal</b>                                                                                                                                                                                                                       |               |         |            |         |      |   |        |               |    |            |
|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------|---------|------------|---------|------|---|--------|---------------|----|------------|
| 2) mount 모드 기동            | \$ <b>tbboot</b><br>Tiberio instance started suspended at MOUNT mode.                                                                                                                                                                                                                                   |               |         |            |         |      |   |        |               |    |            |
| 3) (mount 모드) Datafile 조회 | SQL> <b>SELECT * FROM V\$RECOVER_FILE;</b><br><table><tr><th>FILE#</th><th>ONLINE</th><th>ERROR</th><th>CHANGE#</th><th>TIME</th></tr><tr><td>2</td><td>ONLINE</td><td>open failed</td><td>0</td><td></td></tr></table>                                                                                 | FILE#         | ONLINE  | ERROR      | CHANGE# | TIME | 2 | ONLINE | open failed   | 0  |            |
| FILE#                     | ONLINE                                                                                                                                                                                                                                                                                                  | ERROR         | CHANGE# | TIME       |         |      |   |        |               |    |            |
| 2                         | ONLINE                                                                                                                                                                                                                                                                                                  | open failed   | 0       |            |         |      |   |        |               |    |            |
| 4) (mount 모드) Datafile 생성 | SQL> <b>ALTER DATABASE CREATE DATAFILE 2;</b> <== FILE명 또는 파일번호<br>SQL> <b>SELECT * FROM V\$RECOVER_FILE;</b><br><table><tr><th>FILE#</th><th>ONLINE</th><th>ERROR</th><th>CHANGE#</th><th>TIME</th></tr><tr><td>2</td><td>ONLINE</td><td>file restored</td><td>23</td><td>2019/04/02</td></tr></table> | FILE#         | ONLINE  | ERROR      | CHANGE# | TIME | 2 | ONLINE | file restored | 23 | 2019/04/02 |
| FILE#                     | ONLINE                                                                                                                                                                                                                                                                                                  | ERROR         | CHANGE# | TIME       |         |      |   |        |               |    |            |
| 2                         | ONLINE                                                                                                                                                                                                                                                                                                  | file restored | 23      | 2019/04/02 |         |      |   |        |               |    |            |
| 5) (mount 모드) 복구 명령 수행    | SQL> <b>ALTER DATABASE RECOVER AUTOMATIC DATABASE;</b><br>Database altered.                                                                                                                                                                                                                             |               |         |            |         |      |   |        |               |    |            |
| 6) 데이터베이스 오픈              | \$ <b>tbdown</b><br>\$ <b>tbboot</b><br>Tiberio instance started up (NORMAL mode).                                                                                                                                                                                                                      |               |         |            |         |      |   |        |               |    |            |



# 취소기반 불완전 복구

- 장애 내용 : 일부 Datafile, Archivelog 손상
- 복구 방법 : mount 모드에서 datafile 을 백업 파일로 교체후, archivelog file 을 한 개씩 적용하여 복구를 수행함
- 작업 순서

1) (장애유발) 일부 Datafile 삭제, Archivelog 삭제

2) mount 모드 기동

3) (mount 모드) Datafile 조회

4) (mount 모드) 백업 Datafile 파일 넣기

5) (mount 모드) 복구 명령 수행( 취소기반 )

6) 데이터베이스 오픈( resetlogs 적용 )

# 취소기반 불완전 복구

## 1) (장애유발) Datafile, Archivelog file 삭제

-파일 삭제 후 TIBERO 비정상 종료

### • 데이터 입력(1~7), Archivelog 생성

```
SQL> create table tiber0.t1(c1 number);
```

```
SQL> insert into tiber0.t1 values(1);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(2);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(3);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(4);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(5);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(6);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(7);
commit;
```

```
SQL> SELECT C1
FROM TIBERO.T1;
```

| C1 |
|----|
| 1  |
| 2  |
| 3  |
| 4  |
| 5  |
| 6  |
| 7  |

### • 일부 Datafile 삭제

```
SQL> SELECT NAME FROM V$DATAFILE;
```

```
SQL> !rm /tiber0/tbdata/tiber0/usr001.dtf
```

### • 일부 Archivelog 삭제

```
SQL> COL NAME FOR A45
```

```
SQL> SELECT SEQUENCE#, NAME
FROM V$ARCHIVE_DEST_FILES
ORDER BY SEQUENCE#;
```

최근 아카이브 3건 삭제

```
SQL> !rm /tiber0/tbdata/tiber0/arch/log-t0-r0-s6.arc
!rm /tiber0/tbdata/tiber0/arch/log-t0-r0-s7.arc
!rm /tiber0/tbdata/tiber0/arch/log-t0-r0-s8.arc
```

### • 인스턴스 비정상 종료

```
[tiber0@T1:/tiber0]$ tbdownd abnormal
Tiber0 instance terminated (ABNORMAL mode).
```

# 취소기반 불완전 복구

## 2) mount 모드 기동

-티베로 시작시, MOUNT 모드로 기동됨( 에러메시지 발생 )

```
[tibero@T1:/tibero]$ tbboot
Change core dump dir to /tibero/tibero6/bin/prof.
Listener port = 8629

* Critical Warning : Raise svmode failed. The reason is
* TBR-1024 : Database needs media recovery: open failed(/tibero/tbdata/tibero/usr001.dtf).
* Current server mode is MOUNT.

Tibero 6

TmaxData Corporation Copyright (c) 2008-. All rights reserved.
Tibero instance started suspended at MOUNT mode.
```

# 취소기반 불완전 복구

## 3) (mount 모드) Datafile 조회

- open faild 가 발생한 데이터 파일을 조회함

```
SQL> COL TIME FOR A10
```

```
COL ERROR FOR A30
```

```
SELECT * FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR       | CHANGE# | TIME |
|-------|--------|-------------|---------|------|
| 2     | ONLINE | open failed | 0       |      |

```
SQL> SELECT NAME FROM V$DATAFILE WHERE FILE# = 2;
```

```
NAME
```

```

/tibero/tbdata/tibero/usr001.dtf
```

```
SQL> !ls /tibero/tbdata/tibero/usr001.dtf
```

```
ls: cannot access /tibero/tbdata/tibero/usr001.dtf: No such file or directory
```

# 취소기반 불완전 복구

## 4) (mount 모드) 백업 Datafile 파일 넣기

- 장애 이전에 백업하였던 해당 데이터파일 "usr001.dtf" 을 넣음

```
SQL> !cp /tibero/s/backup/offline/tbdata/tibero/usr001.dtf /tibero/tbdata/tibero
```

- V\$RECOVER\_FILE 조회

```
SQL> COL TIME FOR A10
COL ERROR FOR A30
SELECT FILE#, "ONLINE", ERROR, CHANGE# FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR         | CHANGE# |
|-------|--------|---------------|---------|
| 2     | ONLINE | file restored | 36237   |

# 취소기반 불완전 복구

## 5) (mount 모드) 복구 명령 수행 ( 취소 기반 )

- 복구대상 Datafile 의 Checkpoint TSN 은 36237 이고, 36381 까지 복구가 진행되어야 함을 확인함.

```
SQL> SELECT CKPT_TSN
 FROM V$DATAFILE_HEADER
 WHERE FILE#=2;
```

```
CKPT_TSN

36237
```

```
SQL> SELECT CKPT_TSN
 FROM V$DATAFILE
 WHERE FILE#=2;
```

```
CKPT_TSN

36281
```

- 복구대상 Datafile 의 Checkpoint TSN 은 36237 이므로, log-t0-r0-s3.arc (SEQUENCE# 3)부터 적용되어야 함을 확인함

```
SQL> COL NAME FOR A45
 SELECT NAME, FIRST_CHANGE#, NEXT_CHANGE#-1
 FROM V$ARCHIVE_DEST_FILES ORDER BY SEQUENCE#;
```

| NAME                                          | FIRST_CHANGE# | NEXT_CHANGE#-1 |
|-----------------------------------------------|---------------|----------------|
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s1.arc | 0             | 36207          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s2.arc | 36208         | 36232          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s3.arc | 36233         | 36265          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s4.arc | 36266         | 36269          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s5.arc | 36270         | 36274          |

- 온라인 redolog 가 SEQUENCE# 7 번 부터 시작되므로, 아카이브로그에 6 번이 존재 해야함 ( 장애로 삭제되어 없음 )

```
SQL> SELECT SEQUENCE#, FIRST_CHANGE# FROM V$LOG ORDER BY SEQUENCE#;
```

| SEQUENCE# | FIRST_CHANGE# |
|-----------|---------------|
| 7         | 36280         |
| 8         | 36285         |
| 9         | 36290         |

# 취소기반 불완전 복구

## 5) (mount 모드) 복구 명령 수행 ( 취소 기반 )

- 나머지 모든 Datafile 을 백업 Datafile 로 교체( Datafile 일관성 유지를 위해 )

```
SQL> !cp /tibero/s/backup/offline/tbdata/tibero/*.dtf /tibero/tbdata/tibero
```

- Restore 확인

```
SQL> COL TIME FOR A10
COL ERROR FOR A30
SELECT FILE#, "ONLINE", ERROR, CHANGE# FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR         | CHANGE# |
|-------|--------|---------------|---------|
| 0     | ONLINE | file restored | 36237   |
| 1     | ONLINE | file restored | 36237   |
| 2     | ONLINE | file restored | 36237   |
| 3     | ONLINE | file restored | 36237   |

# 취소기반 불완전 복구

## 5) (mount 모드) 복구 명령 수행 ( 취소 기반 )

### - 취소기반 불완전 복구 시작

```
SQL> ALTER DATABASE RECOVER DATABASE UNTIL CANCEL;
```

### - Archivelog 파일 적용

```
SQL> ALTER DATABASE RECOVER LOGFILE '/tibero/tbdata/tibero/arch/log-t0-r0-s3.arc';
 ALTER DATABASE RECOVER LOGFILE '/tibero/tbdata/tibero/arch/log-t0-r0-s4.arc';
 ALTER DATABASE RECOVER LOGFILE '/tibero/tbdata/tibero/arch/log-t0-r0-s5.arc';
```

### - 취소기반 불완전 복구 종료

```
SQL> ALTER DATABASE RECOVER CANCEL;
```

```
TBR-1074: Incomplete media recovery is finished, but resetlogs is required.
Recovery done until 2019/04/05 13:52:28 TSN 36272.
The last applied log sequence for each redo thread are:
Thread #0 seq 5
```



# 취소기반 불완전 복구

## 6) 데이터베이스 오픈( resetlogs 적용 )

- tbdowm, tbboot resetlogs 명령으로 재기동하여, 인스턴스가 **NORMAL RESETLOGS mode** 로 구동되는지 확인한다.

```
[tibero@T1:/tibero]$ tbdowm
```

```
Tibero instance terminated (NORMAL mode).
```

```
[tibero@T1:/tibero]$ tbboot resetlogs
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
```

```
Listener port = 8629
```

```
Tibero 6
```

```
TmaxData Corporation Copyright (c) 2008-. All rights reserved.
```

```
Tibero instance started up (NORMAL RESETLOGS mode).
```

- 데이터 조회( 복구 결과 )

```
SQL> SELECT * FROM TIBERO.T1;
```

| C1 |
|----|
| 1  |
| 2  |
| 3  |



장애 발생 전

| C1 |
|----|
| 1  |
| 2  |
| 3  |
| 4  |
| 5  |
| 6  |
| 7  |

### \*\* 주의사항

RESETLOGS 부팅 이후에는.  
RESETLOGS 이전에 백업 받은 파일은  
복구작업에 사용할 수 없다. 따라서  
반드시 전체 백업을 받도록 한다.

# 취소기반 불완전 복구

## 6) 데이터베이스 오픈( resetlogs 적용 )

-V\$LOG : REDO 초기화, 다시 SEQUENCE# 1 부터 시작됨

```
SQL> SELECT GROUP#, SEQUENCE#, ARCHIVED
 ,STATUS, FIRST_CHANGE#
 FROM V$LOG
 ORDER BY 1;
```

| GROUP# | SEQUENCE# | ARCHIVED | STATUS  | FIRST_CHANGE# |
|--------|-----------|----------|---------|---------------|
| 0      | -1        | NO       | UNUSED  | 0             |
| 1      | -1        | NO       | UNUSED  | 0             |
| 2      | 1         | NO       | CURRENT | 36275         |

- V\$LOG\_HISTORY : resetlog 이력을 확인 가능

```
SQL> COL RESETLOGS_TIME FOR A20
 SELECT SEQUENCE#, FIRST_CHANGE#
 ,RESETLOGS_CHANGE#
 FROM V$LOG_HISTORY
 ORDER BY NEXT_TIME;
```

| SEQUENCE# | FIRST_CHANGE# | RESETLOGS_CHANGE# |
|-----------|---------------|-------------------|
| 1         | 0             | 0                 |
| 2         | 36208         | 0                 |
| 3         | 36233         | 0                 |
| 4         | 36266         | 0                 |
| 5         | 36270         | 0                 |
| 6         | 36275         | 0                 |
| 7         | 36280         | 0                 |
| 8         | 36285         | 0                 |

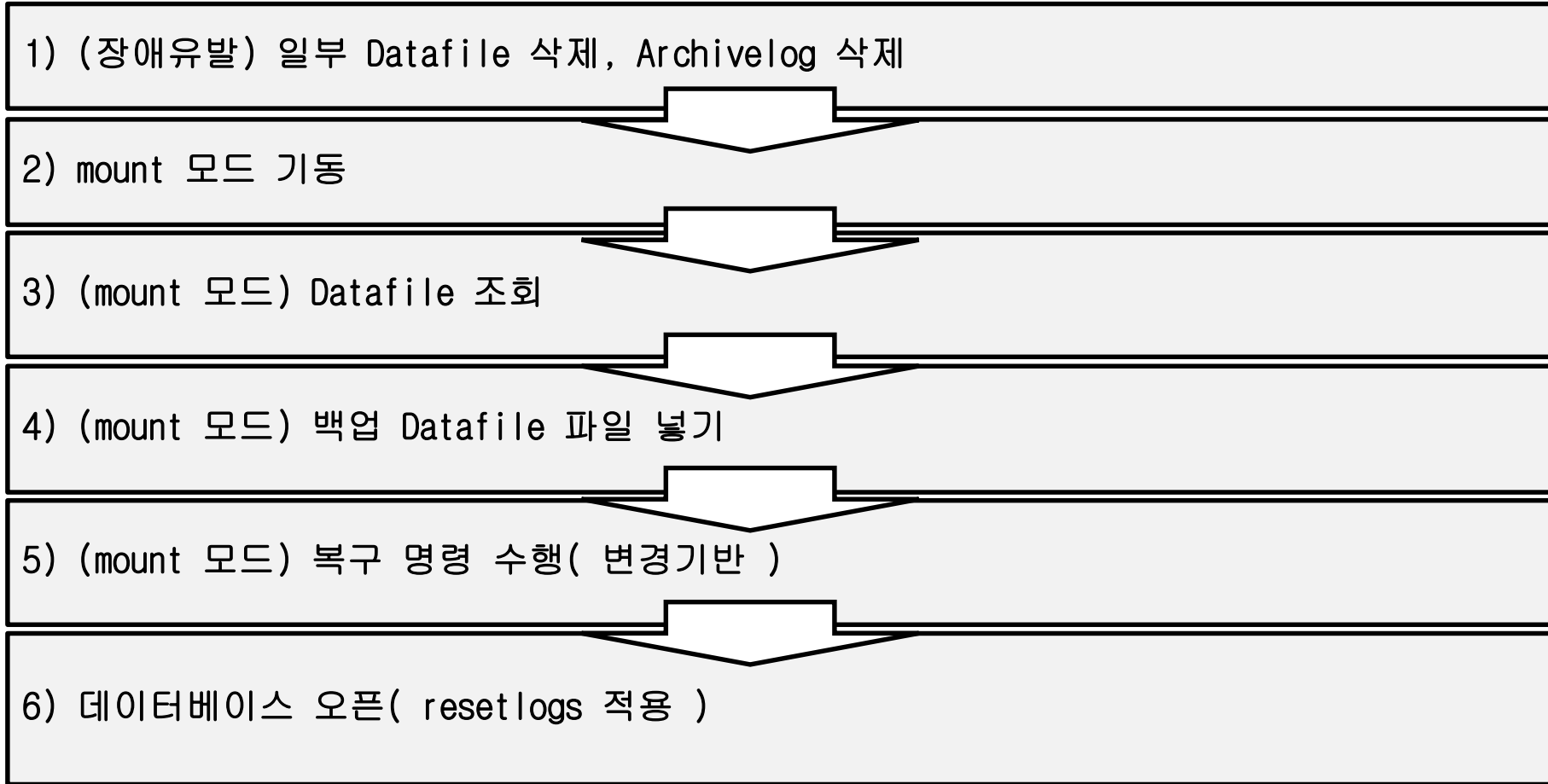
**ALTER SYSTEM SWITCH LOGFILE;**

| GROUP# | SEQUENCE# | ARCHIVED | STATUS   | FIRST_CHANGE# |
|--------|-----------|----------|----------|---------------|
| 0      | 2         | NO       | CURRENT  | 36566         |
| 1      | -1        | NO       | UNUSED   | 0             |
| 2      | 1         | YES      | INACTIVE | 36275         |

| SEQUENCE# | FIRST_CHANGE# | RESETLOGS_CHANGE# |
|-----------|---------------|-------------------|
| 1         | 0             | 0                 |
| 2         | 36208         | 0                 |
| 3         | 36233         | 0                 |
| 4         | 36266         | 0                 |
| 5         | 36270         | 0                 |
| 6         | 36275         | 0                 |
| 7         | 36280         | 0                 |
| 8         | 36285         | 0                 |
| 1         | 36275         | 36275             |

# 변경기반 불완전 복구

- 장애 내용 : 일부 Datafile, Archivelog 손상
- 복구 방법 : mount 모드에서 datafile 을 백업 파일로 교체후, Archivelog 의 TSN 을 이용하여 복구를 수행함
- 작업 순서



# 변경기반 불완전 복구

## 1) (장애유발) Datafile, Archivelog file 삭제

-파일 삭제 후 TIBERO 비정상 종료

### • 데이터 입력(1~7), Archivelog 생성

```
SQL> create table tiber0.t1(c1 number);
```

```
SQL> insert into tiber0.t1 values(1);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(2);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(3);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(4);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(5);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(6);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(7);
commit;
```

```
SQL> SELECT C1
FROM TIBERO.T1;
```

| C1 |
|----|
| 1  |
| 2  |
| 3  |
| 4  |
| 5  |
| 6  |
| 7  |

### • 일부 Datafile 삭제

```
SQL> SELECT NAME FROM V$DATAFILE;
```

```
SQL> !rm /tiber0/tbdata/tiber0/usr001.dtf
```

### • 일부 Archivelog 삭제

```
SQL> COL NAME FOR A45
```

```
SQL> SELECT SEQUENCE#, NAME
FROM V$ARCHIVE_DEST_FILES
ORDER BY SEQUENCE#;
```

최근 아카이브 3건 삭제

```
SQL> !rm /tiber0/tbdata/tiber0/arch/log-t0-r0-s6.arc
!rm /tiber0/tbdata/tiber0/arch/log-t0-r0-s7.arc
!rm /tiber0/tbdata/tiber0/arch/log-t0-r0-s8.arc
```

### • 인스턴스 비정상 종료

```
[tiber0@T1:/tiber0]$ tbdownd abnormal
Tiber0 instance terminated (ABNORMAL mode).
```

# 변경기반 불완전 복구

## 2) mount 모드 기동

-티베로 시작시, MOUNT 모드로 기동됨( 에러메시지 발생 )

```
[tibero@T1:/tibero]$ tbboot
Change core dump dir to /tibero/tibero6/bin/prof.
Listener port = 8629

* Critical Warning : Raise svmode failed. The reason is
* TBR-1024 : Database needs media recovery: open failed(/tibero/tbdata/tibero/usr001.dtf).
* Current server mode is MOUNT.

Tibero 6

TmaxData Corporation Copyright (c) 2008-. All rights reserved.
Tibero instance started suspended at MOUNT mode.
```

# 변경기반 불완전 복구

## 3) (mount 모드) Datafile 조회

- open faild 가 발생한 데이터 파일을 조회함

```
SQL> COL TIME FOR A10
```

```
COL ERROR FOR A30
```

```
SELECT * FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR       | CHANGE# | TIME |
|-------|--------|-------------|---------|------|
| 2     | ONLINE | open failed | 0       |      |

```
SQL> SELECT NAME FROM V$DATAFILE WHERE FILE# = 2;
```

```
NAME
```

```
/tibero/tbdata/tibero/usr001.dtf
```

```
SQL> !ls /tibero/tbdata/tibero/usr001.dtf
```

```
ls: cannot access /tibero/tbdata/tibero/usr001.dtf: No such file or directory
```

# 변경기반 불완전 복구

## 4) (mount 모드) 백업 Datafile 파일 넣기

- 장애 이전에 백업하였던 해당 데이터파일 "usr001.dtf" 을 넣음

```
SQL> !cp /tibero/s/backup/offline/tbdata/tibero/usr001.dtf /tibero/tbdata/tibero
```

- V\$RECOVER\_FILE 조회

```
SQL> COL TIME FOR A10
COL ERROR FOR A30
SELECT FILE#, "ONLINE", ERROR, CHANGE# FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR         | CHANGE# |
|-------|--------|---------------|---------|
| 2     | ONLINE | file restored | 36237   |

# 변경기반 불완전 복구

## 5) (mount 모드) 복구 명령 수행 ( 변경 기반 )

- 복구대상 Datafile 의 Checkpoint TSN 은 36237 이고, 36381 까지 복구가 진행되어야 함을 확인함.

```
SQL> SELECT CKPT_TSN
 FROM V$DATAFILE_HEADER
 WHERE FILE#=2;
```

```
CKPT_TSN

36237
```

```
SQL> SELECT CKPT_TSN
 FROM V$DATAFILE
 WHERE FILE#=2;
```

```
CKPT_TSN

36281
```

- 복구대상 Datafile 의 Checkpoint TSN 은 36237 이므로, log-t0-r0-s3.arc (SEQUENCE# 3)부터 적용되어야 함을 확인함

```
SQL> COL NAME FOR A45
 SELECT NAME, FIRST_CHANGE#, NEXT_CHANGE#-1
 FROM V$ARCHIVE_DEST_FILES ORDER BY SEQUENCE#;
```

| NAME                                          | FIRST_CHANGE# | NEXT_CHANGE#-1 |
|-----------------------------------------------|---------------|----------------|
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s1.arc | 0             | 36207          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s2.arc | 36208         | 36232          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s3.arc | 36233         | 36265          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s4.arc | 36266         | 36269          |
| /tiberio/tbdata/tiberio/arch/log-t0-r0-s5.arc | 36270         | 36274          |

- 온라인 redolog 가 SEQUENCE# 7 번 부터 시작되므로, 아카이브로그에 6 번이 존재 해야함 ( 장애로 삭제되어 없음 )

```
SQL> SELECT SEQUENCE#, FIRST_CHANGE# FROM V$LOG ORDER BY SEQUENCE#;
```

| SEQUENCE# | FIRST_CHANGE# |
|-----------|---------------|
| 7         | 36280         |
| 8         | 36285         |
| 9         | 36290         |



# 변경기반 불완전 복구

## 5) (mount 모드) 복구 명령 수행 ( 취소 기반 )

- 나머지 모든 Datafile 을 백업 Datafile 로 교체( Datafile 일관성 유지를 위해 )

```
SQL> !cp /tibero/s/backup/offline/tbdata/tibero/*.dtf /tibero/tbdata/tibero
```

- Restore 확인

```
SQL> COL TIME FOR A10
COL ERROR FOR A30
SELECT FILE#, "ONLINE", ERROR, CHANGE# FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR         | CHANGE# |
|-------|--------|---------------|---------|
| 0     | ONLINE | file restored | 36237   |
| 1     | ONLINE | file restored | 36237   |
| 2     | ONLINE | file restored | 36237   |
| 3     | ONLINE | file restored | 36237   |

# 변경기반 불완전 복구

## 5) (mount 모드) 복구 명령 수행 ( 취소 기반 )

- 변경기반 불완전 복구
- ALTER DATABASE RECOVER AUTOMATIC DATABASE UNTIL CHANGE [ 복구할 TSN 번호 지정 ] ;

```
SQL> ALTER DATABASE RECOVER AUTOMATIC DATABASE UNTIL CHANGE 36274;
TBR-1070: Incomplete media recovery is finished, but resetlogs is required.
```

# 변경기반 불완전 복구

## 6) 데이터베이스 오픈( resetlogs 적용 )

- tbdownd, tbboot resetlogs 명령으로 재기동하여, 인스턴스가 **NORMAL RESETLOGS mode** 로 구동되는지 확인한다.

```
[tibero@T1:/tibero]$ tbdownd
```

```
Tibero instance terminated (NORMAL mode).
```

```
[tibero@T1:/tibero]$ tbboot resetlogs
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
```

```
Listener port = 8629
```

```
Tibero 6
```

```
TmaxData Corporation Copyright (c) 2008-. All rights reserved.
```

```
Tibero instance started up (NORMAL RESETLOGS mode).
```

- 데이터 조회( 복구 결과 )

```
SQL> SELECT * FROM TIBERO.T1;
```

| C1 |
|----|
| 1  |
| 2  |
| 3  |



장애 발생 전

| C1 |
|----|
| 1  |
| 2  |
| 3  |
| 4  |
| 5  |
| 6  |
| 7  |

### \*\* 주의사항

RESETLOGS 부팅 이후에는.  
RESETLOGS 이전에 백업 받은 파일은  
복구작업에 사용할 수 없다. 따라서  
반드시 전체 백업을 받도록 한다.

# 변경기반 불완전 복구

## 6) 데이터베이스 오픈( resetlogs 적용 )

-V\$LOG : REDO 초기화, 다시 SEQUENCE# 1 부터 시작됨

```
SQL> SELECT GROUP#, SEQUENCE#, ARCHIVED
 ,STATUS, FIRST_CHANGE#
 FROM V$LOG
 ORDER BY 1;
```

| GROUP# | SEQUENCE# | ARCHIVED | STATUS  | FIRST_CHANGE# |
|--------|-----------|----------|---------|---------------|
| 0      | -1        | NO       | UNUSED  | 0             |
| 1      | -1        | NO       | UNUSED  | 0             |
| 2      | 1         | NO       | CURRENT | 36273         |

- V\$LOG\_HISTORY : resetlog 이력을 확인 가능

```
SQL> COL RESETLOGS_TIME FOR A20
 SELECT SEQUENCE#, FIRST_CHANGE#
 ,RESETLOGS_CHANGE#
 FROM V$LOG_HISTORY
 ORDER BY NEXT_TIME;
```

| SEQUENCE# | FIRST_CHANGE# | RESETLOGS_CHANGE# |
|-----------|---------------|-------------------|
| 1         | 0             | 0                 |
| 2         | 36208         | 0                 |
| 3         | 36233         | 0                 |
| 4         | 36266         | 0                 |
| 5         | 36270         | 0                 |
| 6         | 36275         | 0                 |
| 7         | 36280         | 0                 |
| 8         | 36285         | 0                 |

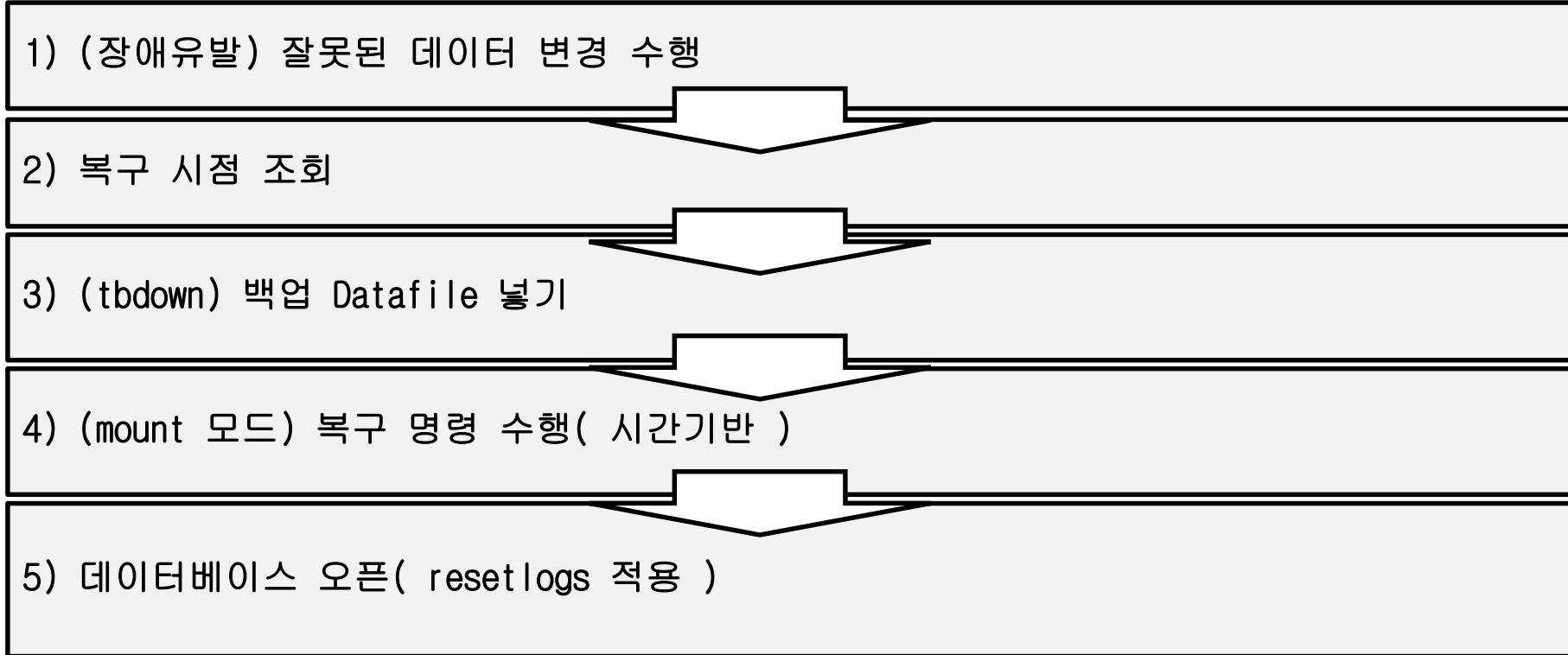
**ALTER SYSTEM SWITCH LOGFILE;**

| GROUP# | SEQUENCE# | ARCHIVED | STATUS   | FIRST_CHANGE# |
|--------|-----------|----------|----------|---------------|
| 0      | 2         | NO       | CURRENT  | 36574         |
| 1      | -1        | NO       | UNUSED   | 0             |
| 2      | 1         | YES      | INACTIVE | 36273         |

| SEQUENCE# | FIRST_CHANGE# | RESETLOGS_CHANGE# |
|-----------|---------------|-------------------|
| 1         | 0             | 0                 |
| 2         | 36208         | 0                 |
| 3         | 36233         | 0                 |
| 4         | 36266         | 0                 |
| 5         | 36270         | 0                 |
| 6         | 36275         | 0                 |
| 7         | 36280         | 0                 |
| 8         | 36285         | 0                 |
| 1         | 36273         | 36273             |

# 시간기반 불완전 복구

- 장애 내용 : 잘못된 데이터 변경 수행
- 복구 방법 : mount 모드에서 datafile 을 백업 파일로 교체후, 시점을 적용하여 복구를 수행함
- 작업 순서



# 시간기반 불완전 복구

## 1) (장애유발) 잘못된 데이터 변경 수행

-데이터 변경 실행 후

### • 데이터 입력(1~7), ArchiveLog

```
SQL> create table tiber0.t1(c1 number);
```

```
SQL> insert into tiber0.t1 values(1);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(2);
commit;
```

```
alter system switch logfile;
```

```
insert into tiber0.t1 values(3);
commit;
```

```
alter system switch logfile;
insert into tiber0.t1 values(4);
commit;
```

```
alter system switch logfile;
insert into tiber0.t1 values(5);
commit;
```

```
alter system switch logfile;
insert into tiber0.t1 values(6);
commit;
```

```
alter system switch logfile;
insert into tiber0.t1 values(7);
commit;
```

```
alter system switch logfile;
insert into tiber0.t1 values(7);
commit;
```

```
alter system switch logfile;
insert into tiber0.t1 values(7);
commit;
```

```
SELECT C1
FROM TIBERO.T1;
```

```

C1
```

```
1
```

```
2
```

```
3
```

```
4
```

```
5
```

```
6
```

```
7
```

```
(4);
```

```
(5);
```

```
(6);
```

### • 데이터 처리 작업 등록

#### - JOB 등록

```
SQL> ALTER SYSTEM SET JOB_HISTORY_LOG=Y;
```

```
SQL> DECLARE
 JOB NUMBER;
BEGIN
 DBMS_JOB_WITH_NAME.SUBMIT
 (JOB
 ,NAME=>'EDU_JOB01'
 ,WHAT=>'DELETE TIBERO.T1;'
 ,NEXT_DATE=>SYSDATE+(1/24/60)
 ,INTERVAL=>'SYSDATE+1');
END;
/
```

#### - 등록된 JOB 조회

```
SQL> SET LINESIZE 120
```

```
COL NAME FOR A10
```

```
COL INTERVAL FOR A40
```

```
COL NEXT_DATE FOR A20
```

```
SELECT NAME
```

```
 ,TO_CHAR(NEXT_DATE, 'YYYY/MM/DD HH24:MI:SS') NEXT_DATE
```

```
 ,INTERVAL
```

```
FROM DBA_JOBS_WITH_NAME;
```

```
NAME NEXT_DATE INTERVAL

```

```
EDU_JOB01 2019/04/04 18:19:54 SYSDATE+1
```

### • 잘못된 데이터 변경 수행

#### - JOB 수행 결과

```
SQL> SELECT C1 FROM TIBERO.T1;
```

```
0 row selected.
```

# 시간기반 불완전 복구

## 2) 복구 시점 조회

- 잘못된 작업이 수행된 시점을 확인하기

```
SQL> COL NAME FOR A10
COL START_DATE FOR A20
COL END_DATE FOR A20
COL RESULT FOR A10
SELECT NAME
 ,TO_CHAR(START_DATE,'YYYY/MM/DD HH24:MI:SS') START_DATE
 ,TO_CHAR(END_DATE,'YYYY/MM/DD HH24:MI:SS') END_DATE
 ,RESULT
 FROM DBA_JOB_HISTORY
 WHERE NAME = 'EDU_JOB01'
 ORDER BY START_DATE;
```

| NAME      | START_DATE          | END_DATE            | RESULT  |
|-----------|---------------------|---------------------|---------|
| EDU_JOB01 | 2019/04/04 18:19:54 | 2019/04/04 18:19:54 | SUCCESS |

# 시간기반 불완전 복구

## 3) (tbdown) 백업 Datafile 넣기

- 인스턴스 종료

```
$ tbdown immediate
```

- 모든 Datafile 을 백업 Datafile 로 교체( Datafile 일관성 유지를 위해 )

```
$ cp /tibero/s/backup/offline/tbdata/tibero/*.dtf /tibero/tbdata/tibero
```

- mount 모드 기동

```
$ tbboot mount
```

- Restore 확인

```
SQL> COL TIME FOR A10
SQL> COL ERROR FOR A30
SQL> SELECT * FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR         | CHANGE# | TIME       |
|-------|--------|---------------|---------|------------|
| 0     | ONLINE | file restored | 36237   | 2019/04/04 |
| 1     | ONLINE | file restored | 36237   | 2019/04/04 |
| 2     | ONLINE | file restored | 36237   | 2019/04/04 |
| 3     | ONLINE | file restored | 36237   | 2019/04/04 |



# 시간기반 불완전 복구

## 4) (mount 모드) 복구 명령 수행( 시간기반 )

- 시간 포맷을 설정

```
SQL> ALTER SYSTEM SET NLS_DATE_FORMAT='YYYY/MM/DD HH24:MI:SS';
```

- 앞서 확인한 복구 시점을 적용하여 복구 명령 실행

- ALTER DATABASE RECOVER AUTOMATIC DATABASE UNTIL TIME [복구할 시점 지정]

```
SQL> ALTER DATABASE RECOVER AUTOMATIC DATABASE UNTIL TIME '2019/04/04 18:19:53';
```

```
TBR-1070: Incomplete media recovery is finished, but resetlogs is required.
```

# 시간기반 불완전 복구

## 5) 데이터베이스 오픈( resetlogs 적용 )

- tbdowm, tbboot resetlogs 명령으로 재기동하여, 인스턴스가 **NORMAL RESETLOGS mode** 로 구동되는지 확인한다.

```
[tibero@T1:/tibero]$ tbdowm
```

```
Tibero instance terminated (NORMAL mode).
```

```
[tibero@T1:/tibero]$ tbboot resetlogs
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
```

```
Listener port = 8629
```

```
Tibero 6
```

```
TmaxData Corporation Copyright (c) 2008-. All rights reserved.
```

```
Tibero instance started up (NORMAL RESETLOGS mode).
```

- 데이터 조회( 복구 결과 )

```
SQL> SELECT * FROM TIBERO.T1;
```

C1

-----

1  
2  
3  
4  
5  
6  
7

**\*\* 복구 작업 이전**

SQL> SELECT \* FROM TIBERO.T1;

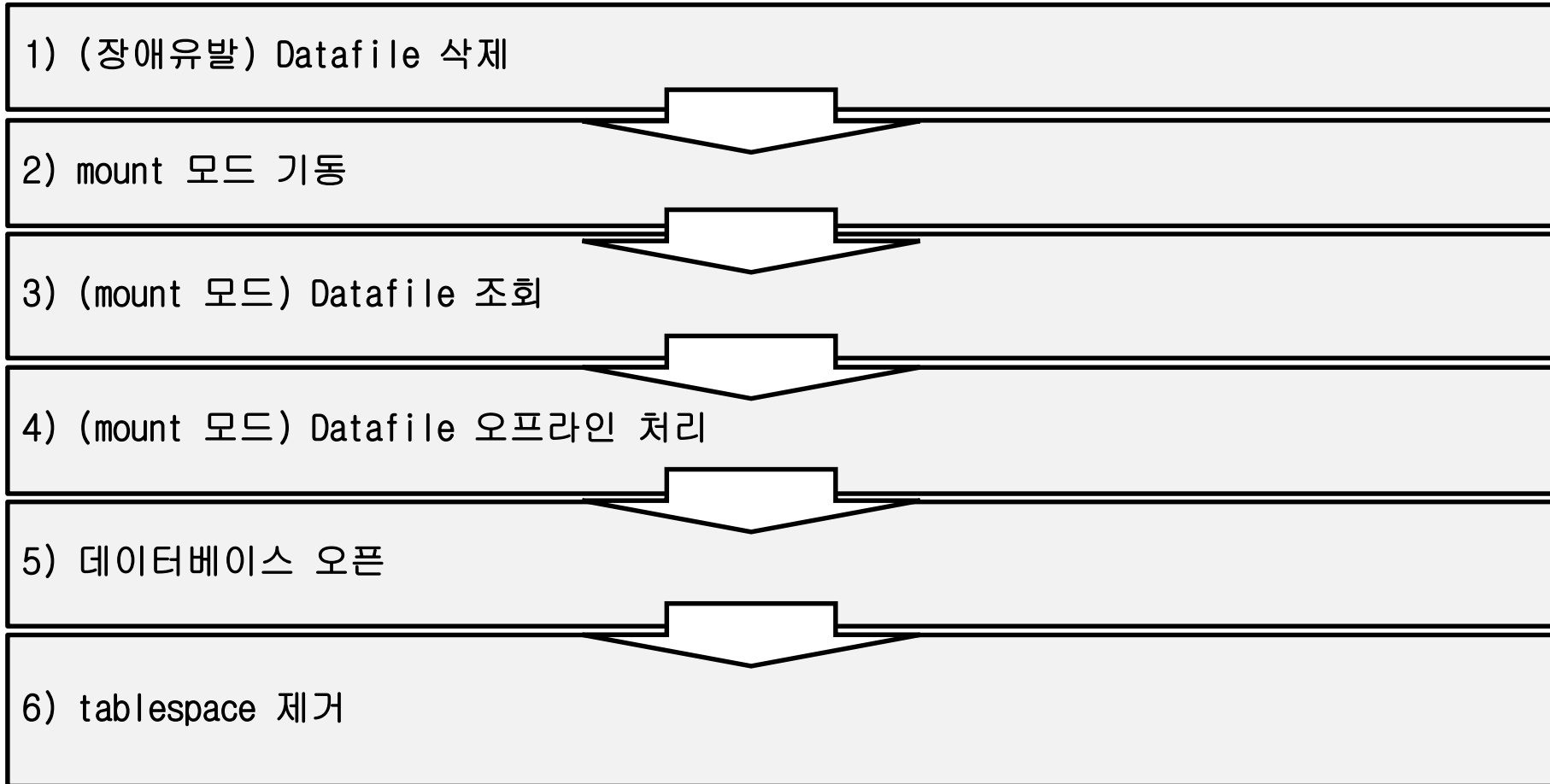
0 row selected.

### \*\* 주의사항

- RESETLOGS 부팅 이후, 전체 백업 수행 필요함  
( 향후 복구작업에서, RESETLOGS 이전의 백업은  
사용 불가함.)

# 백업 파일이 없는 datafile 제거

- 장애 내용 : Datafile 손상 (백업 Datafile 없음)
- 복구 방법 : mount 모드에서 datafile 오프라인 수행, 데이터베이스 open 이후 tablespace 삭제처리 함
- 작업 순서



# 백업 파일이 없는 datafile 제거

## 1) (장애유발) Datafile 삭제

-Datafile을 삭제한 후 TIBERO 비정상 종료

```
SQL> SELECT NAME FROM V$DATAFILE;
```

```
NAME
```

```

/tibero/tbdata/tibero/system001.dtf
/tibero/tbdata/tibero/undo001.dtf
/tibero/tbdata/tibero/usr001.dtf
/tibero/tbdata/tibero/tpr_ts.dtf
/tibero/tbdata/tibero/edu01.dtf
```

```
SQL> !rm /tibero/tbdata/tibero/edu01.dtf
```

```
SQL> quit
```

```
Disconnected.
```

```
[tibero@T1:/tibero]$ tbdwn abnormal
```

```
Tibero instance terminated (ABNORMAL mode).
```

# 백업 파일이 없는 datafile 제거

## 2) mount 모드 기동

- 티베로 시작시, MOUNT 모드로 기동됨( 에러메시지 발생 )

```
[tibero@T1:/tibero]$ tbboot
Change core dump dir to /tibero/tibero6/bin/prof.
Listener port = 8629

* Critical Warning : Raise svmode failed. The reason is
* TBR-1024 : Database needs media recovery: open failed(/tibero/tbdata/tibero/edu01.dtf).
* Current server mode is MOUNT.

Tibero 6

TmaxData Corporation Copyright (c) 2008-. All rights reserved.
Tibero instance started suspended at MOUNT mode.
```

# 백업 파일이 없는 datafile 제거

## 3) (mount 모드) 데이터 파일 조회

- open faild 가 발생한 데이터 파일을 조회함

```
SQL> COL TIME FOR A10
```

```
SQL> COL ERROR FOR A30
```

```
SQL> SELECT * FROM V$RECOVER_FILE;
```

| FILE# | ONLINE | ERROR       | CHANGE# | TIME |
|-------|--------|-------------|---------|------|
| 4     | ONLINE | open failed | 0       |      |

```
SQL> SELECT NAME FROM V$DATAFILE WHERE FILE# = 4;
```

```
NAME
```

```

/tibero/tbdata/tibero/edu01.dtf
```

```
SQL> !ls /tibero/tbdata/tibero/edu01.dtf
```

```
ls: cannot access /tibero/tbdata/tibero/edu01.dtf: No such file or directory
```

# 백업 파일이 없는 datafile 제거

## 4) (mount 모드) Datafile 오프라인 처리

- 장애 Datafile 의 Tablespace 확인

```
SQL> COL "TABLESPACE" FOR A15
 COL "DATAFILE" FOR A35
 SET LINESIZE 100
 SELECT T.NAME "TABLESPACE", DD.NAME "DATAFILE", DD.FILE#, DD.STATUS, DD.ENABLED
 FROM V$TABLESPACE T, V$DATAFILE D, V$DATAFILE DD
 WHERE T.TS#= D.TS# AND D.FILE#=4 AND T.TS# = DD.TS#;
```

| TABLESPACE | DATAFILE                        | FILE# | STATUS | ENABLED    |
|------------|---------------------------------|-------|--------|------------|
| USR        | /tibero/tbdata/tibero/edu01.dtf | 4     | ONLINE | READ WRITE |

- 장애 Datafile 오프라인 수행

```
SQL> ALTER DATABASE DATAFILE '/tibero/tbdata/tibero/edu02.dtf' OFFLINE FOR DROP;
```

- 장애 Datafile 오프라인 수행 확인

```
SQL> SELECT T.NAME "TABLESPACE", DD.NAME "DATAFILE", DD.FILE#, DD.STATUS, DD.ENABLED
 FROM V$TABLESPACE T, V$DATAFILE D, V$DATAFILE DD
 WHERE T.TS#= D.TS# AND D.FILE#=4 AND T.TS# = DD.TS#;
```

| TABLESPACE | DATAFILE                        | FILE# | STATUS  | ENABLED  |
|------------|---------------------------------|-------|---------|----------|
| USR        | /tibero/tbdata/tibero/edu01.dtf | 4     | RECOVER | DISABLED |

# 백업 파일이 없는 datafile 제거

## 5) 데이터베이스 오픈

- 장애 Datafile 을 오프라인 처리 하였으므로, 정상 기동 됨

```
[tibero@T1:/tibero]$ tbdown immediate
```

```
Tibero instance terminated (IMMEDIATE mode).
```

```
[tibero@T1:/tibero]$ tbboot
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
```

```
Listener port = 8629
```

```
Tibero 6
```

```
TmaxData Corporation Copyright (c) 2008-. All rights reserved.
```

```
Tibero instance started up (NORMAL mode).
```



# 백업 파일이 없는 datafile 제거

## 6) Tablespace 제거

- 장애 Datafile 의 Tablespace 는 사용불능 이므로, 삭제 처리 한다.

```
SQL> COL TABLESPACE_NAME FOR A15
SQL> SELECT TABLESPACE_NAME, TS_ID, STATUS FROM DBA_TABLESPACES WHERE TABLESPACE_NAME = 'EDU';
```

| TABLESPACE_NAME | TS_ID | STATUS |
|-----------------|-------|--------|
| EDU             | 5     | ONLINE |

```
SQL> SELECT FILE_ID, STATUS FROM DBA_DATA_FILES WHERE TABLESPACE_NAME='EDU';

0 row selected.

SQL> DROP TABLESPACE EDU INCLUDING CONTENTS AND DATAFILES;

Tablespace 'EDU' dropped.
```

# 복구 관리자(RMGR) 를 이용한 백업/복구

- RMGR은 다양한 백업/복구 시나리오를 지원하도록 구성되어 있다.

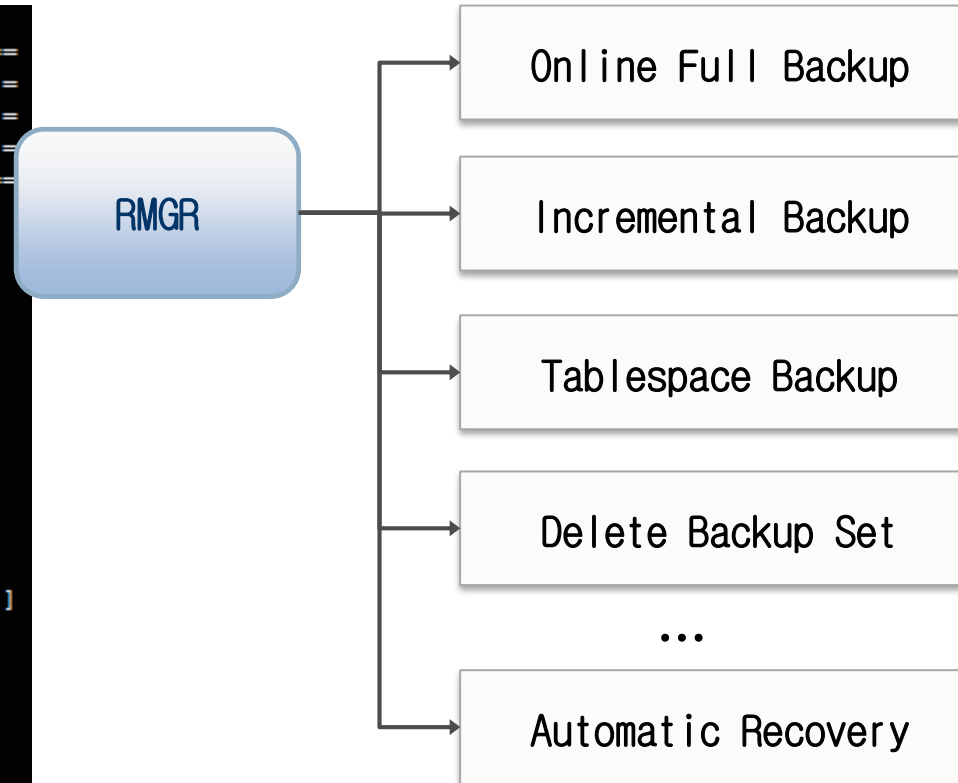
```
[tibero@T1:/tibero]$ tbrmgr
=====
= Recovery Manager(RMGR) starts =
= =
= TmaxData Corporation Copyright (c) 2008-. All rights reserved. =
=====

usage:
 tbrmgr backup [OPTIONS]
 [{-i|--incremental, -C|--cumulative}]
 [-u|--skip-unused]
 [--tablespace TS_NAME1[,TS_NAME2...]]
 [--overwrite]
 [--with-archivelog]
 [--with-password-file]
 [-o BACKUP_DIRECTORY]
 [[-p|--parallel] THREAD_COUNT]

 tbrmgr recover [OPTIONS]
 [--untiltime YYYYMMDDHHmmss, --untilchange TSN]
 [--tablespace TS_NAME1[,TS_NAME2...]]
 [--with-archivelog]
 [--with-password-file]
 [--recover-to]
 [-o BACKUP_DIRECTORY]
 [[-p|--parallel] THREAD_COUNT]

 tbrmgr delete [OPTIONS]
 [--backup_set BACKUP_SET_ID]
 [--beforetime YYYYMMDDHHmmss]
 [-o BACKUP_DIRECTORY]

OPTIONS:
 --userid USERID[/PASSWD] [@SID]
 --help Print this message and exit.
```



## RMGR Backup 정보 조회

- V\$BACKUP\_SET : Backup Set 정보를 조회
- V\$BACKUP\_ARCHIVED\_LOG : Archive Log Backup 정보 조회
- \$BACKUP\_SET\_TABLESPACE : Backup Set 의 Tablespace 정보 조회

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자 기능

| 구분                      | 설명                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Online Full Backup      | <ul style="list-style-type: none"><li>➤ <b>Tibero</b> 데이터베이스에 속한 전체 데이터 파일을 온라인 백업한다.</li><li>➤ 온라인 백업을 위해서는 데이터베이스가 <b>ARCHIVELOG</b> 모드이어야 한다.</li><li>➤ 처리 순서<ol style="list-style-type: none"><li>1) <b>BEGIN BACKUP</b><ul style="list-style-type: none"><li>- RMGR은 자동으로 데이터베이스의 <b>Begin Backup</b> 기능을 사용하여 모든 테이블스페이스를 <b>Hot Backup</b> 상태로 만들고 백업을 진행한다. 이때 백업할 데이터파일을 <b>v\$datafile</b> 을 조회하여 자동으로 결정해 준다.</li></ul></li><li>2) <b>END BACKUP</b><ul style="list-style-type: none"><li>- 백업을 완료하면 데이터베이스의 <b>End Backup</b> 기능을 사용하여 모든 테이블스페이스를 <b>Hot Backup</b> 상태로부터 해제한다.</li></ul></li></ol></li></ul> |
| Incremental Full Backup | <ul style="list-style-type: none"><li>➤ RMGR 을 통해 온라인 백업을 받았으면 이를 이용하여 <b>Incremental Backup</b> 을 할 수 있다.</li><li>➤ <b>Incremental Backup</b> 이란 백업을 받을 때 전체 파일을 받는 것이 아니라 이전 백업과의 차이만을 기록하는 방식으로 백업에 소모되는 디스크 공간을 획기적으로 줄일 수 있다.</li><li>➤ <b>Incremental Backup</b> 은 RMGR 을 통해서만 제공되는 기능으로, 이 기능을 사용 하려면 이전에 RMGR 을 통해 <b>Online Full Backup</b> 을 받았어야 한다.</li></ul>                                                                                                                                                                                                                                                  |
| Automatic Recovery      | <ul style="list-style-type: none"><li>➤ RMGR 로 만들어진 백업본을 이용하여 자동 <b>Recovery</b> 를 진행하며, 이를 위해서는 백업을 만들 당시에 생성된 RMGR <b>info file</b> 이 필요하다. <b>Online Full Backup / Incremental Full Backup</b> 정보를 분석하여 자동으로 <b>Merge</b> 후 <b>Recovery</b> 해준다.</li></ul>                                                                                                                                                                                                                                                                                                                                                              |

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자 정보 파일

- RMGR 로 백업을 진행하면 RMGR Info 라는 텍스트 파일이 생성된다.
- RMGR Info 파일에는 백업정보가 담겨져 있으며, Incremental Full Backup 시에 사용이 된다.
- RMGR Info 파일을 분실하게 되면, 백업본을 이용한 복구를 할 수 없으니 주의해야 한다.

## ● 복구 관리자 옵션

- RMGR 은 Shell Command 로 실행되며, 다음과 같은 옵션을 사용할 수 있다.

| 옵션                       | 설명                                                                  |
|--------------------------|---------------------------------------------------------------------|
| <b>backup</b>            | RMGR 을 통해 백업을 진행한다.                                                 |
| <b>recovery</b>          | RMGR 로 받아놓은 백업본을 이용하여 복구를 진행한다.                                     |
| <b>-v, --verbose</b>     | RMGR 의 진행상황을 자세히 출력해준다.                                             |
| <b>-h, --help</b>        | RMGR 의 옵션 사용법을 출력해준다.                                               |
| <b>-i, --incremental</b> | Incremental Full Backup 을 한다.                                       |
| <b>-b</b>                | RMGR info 파일의 위치를 지정한다. RMGR info 파일은 이전에 RMGR을 통해 백업받은 디렉터리에 생성된다. |
| <b>-o</b>                | 백업받을 디렉터리를 지정한다.                                                    |

# 복구 관리자(RMGR) 를 이용한 백업/복구

- 복구 관리자를 이용한 백업/복구 예제

● Online Full Backup

● Incremental Backup : 가장 최신 백업에 대한 Incremental backup(\*\* -i, incremental 적용 )

● Incremental Backup : 가장 최신 백업에 대한 Incremental backup(\*\* -i, incremental 적용 )

● Incremental Backup : 마지막 full backup에 대한 incremental backup(\*\* -C, cumulative 적용 )

● Incremental Backup : 가장 최신 백업에 대한 Incremental backup(\*\* -i, incremental 적용 )

● 백업정보 조회

● Automatic Recovery

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자를 이용한 백업/복구 예제 - Online Full Backup

```
[tibero@T1:/tibero]$ tbrmgr backup -c -u --with-archivelog -o /tibero/s/backup/rmgr
```

```
=====
= Recovery Manager(RMGR) starts =
= =
= TmaxData Corporation Copyright (c) 2008-. All rights reserved. =
=====
RMGR - ONLINE backup
=====
DB connected
archive log check succeeded
BACKUP (set_id:1, ts_id:0, df_id:0)
 100.00% |=====>| 12800/12800 blks 1.12s
Synchronizing...
BACKUP (set_id:1, ts_id:1, df_id:1)
 100.00% |=====>| 25600/25600 blks 2.26s
Synchronizing...
BACKUP (set_id:1, ts_id:3, df_id:2)
 100.00% |=====>| 12800/12800 blks 0.00s
Synchronizing...
BACKUP (set_id:1, ts_id:4, df_id:3)
 100.00% |=====>| 1280/1280 blks 0.07s
Synchronizing...

Switching logfiles...Backing up archivelog filesArchivelog log-t0-r0-s3.arc (thread: 0, low_tsn: 36233,
next_tsn: 36359) copied
Database full backup succeeded
DB disconnected
RMGR backup ends
```

Backup set 1

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자를 이용한 백업/복구 예제 - Incremental Backup

```
[tibero@T1:/tibero]$ tbrmgr backup -i -c -u --with-archivelog -o /tibero/s/backup/rmgr
```

```
=====
= Recovery Manager(RMGR) starts =
= =
= TmaxData Corporation Copyright (c) 2008-. All rights reserved. =
=====
```

Backup set 2

```
=====
RMGR - INCREMENTAL backup
=====
```

```
DB connected
```

```
archive log check succeeded
```

```
BACKUP (set_id:2, ts_id:0, df_id:0)
```

```
100.00% |=====>| 12800/12800 blks 0.00s
```

```
Synchronizing...
```

```
BACKUP (set_id:2, ts_id:1, df_id:1)
```

```
100.00% |=====>| 25600/25600 blks 0.02s
```

```
Synchronizing...
```

```
BACKUP (set_id:2, ts_id:3, df_id:2)
```

```
100.00% |=====>| 12800/12800 blks 0.00s
```

```
Synchronizing...
```

```
BACKUP (set_id:2, ts_id:4, df_id:3)
```

```
100.00% |=====>| 1280/1280 blks 0.00s
```

```
Synchronizing...
```

```
Switching logfiles...Backing up archivelog filesArchivelog log-t0-r0-s7.arc (thread: 0, low_tsn: 36366, next_tsn: 36404) copied
```

```
Database incremental backup succeeded
```

```
DB disconnected
```

```
RMGR backup ends
```

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자를 이용한 백업/복구 예제 - Incremental Backup

```
[tibero@T1:/tibero]$ tbrmgr backup -i -c -u --with-archivelog -o /tibero/s/backup/rmgr
```

```
=====
= Recovery Manager(RMGR) starts =
= =
= TmaxData Corporation Copyright (c) 2008-. All rights reserved. =
=====
```

Backup set 3

```
=====
RMGR - INCREMENTAL backup
=====
```

```
DB connected
```

```
archive log check succeeded
```

```
BACKUP (set_id:3, ts_id:0, df_id:0)
```

```
100.00% |=====>| 12800/12800 blks 0.02s
```

```
Synchronizing...
```

```
BACKUP (set_id:3, ts_id:1, df_id:1)
```

```
100.00% |=====>| 25600/25600 blks 0.02s
```

```
Synchronizing...
```

```
BACKUP (set_id:3, ts_id:3, df_id:2)
```

```
100.00% |=====>| 12800/12800 blks 0.02s
```

```
Synchronizing...
```

```
BACKUP (set_id:3, ts_id:4, df_id:3)
```

```
100.00% |=====>| 1280/1280 blks 0.00s
```

```
Synchronizing...
```

```
Switching logfiles...Backing up archivelog filesArchivelog log-t0-r0-s11.arc (thread: 0, low_tsn: 36411, next_tsn: 36434) copied
```

```
Database incremental backup succeeded
```

```
DB disconnected
```

```
RMGR backup ends
```



# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자를 이용한 백업/복구 예제 - Incremental Backup

```
[tibero@T1:/tibero]$ tbrmgr backup -C -c -u --with-archivelog -o /tibero/s/backup/rmgr
```

```
=====
= Recovery Manager(RMGR) starts =
= =
= TmaxData Corporation Copyright (c) 2008-. All rights reserved. =
=====
```

Backup set 4

```
=====
RMGR - CUMULATIVE INCREMENTAL backup
=====
```

DB connected

archive log check succeeded

BACKUP (set\_id:4, ts\_id:0, df\_id:0)

100.00% |=====>| 12800/12800 blks 0.02s

Synchronizing...

BACKUP (set\_id:4, ts\_id:1, df\_id:1)

100.00% |=====>| 25600/25600 blks 0.02s

Synchronizing...

BACKUP (set\_id:4, ts\_id:3, df\_id:2)

100.00% |=====>| 12800/12800 blks 0.00s

Synchronizing...

BACKUP (set\_id:4, ts\_id:4, df\_id:3)

100.00% |=====>| 1280/1280 blks 0.00s

Synchronizing...

Switching logfiles...Backing up archivelog filesArchivelog log-t0-r0-s15.arc (thread: 0, low\_tsn: 36441, next\_tsn: 36467) copied

Database incremental backup succeeded

DB disconnected

RMGR backup ends

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자를 이용한 백업/복구 예제 - Incremental Backup

```
[tibero@T1:/tibero]$ tbrmgr backup -i -c -u --with-archivelog -o /tibero/s/backup/rmgr
```

```
=====
= Recovery Manager(RMGR) starts =
= =
= TmaxData Corporation Copyright (c) 2008-. All rights reserved. =
=====
RMGR - INCREMENTAL backup
=====
DB connected
archive log check succeeded
BACKUP (set_id:5, ts_id:0, df_id:0)
 100.00% |=====>| 12800/12800 blks 0.00s
Synchronizing...
BACKUP (set_id:5, ts_id:1, df_id:1)
 100.00% |=====>| 25600/25600 blks 0.02s
Synchronizing...
BACKUP (set_id:5, ts_id:3, df_id:2)
 100.00% |=====>| 12800/12800 blks 0.00s
Synchronizing...
BACKUP (set_id:5, ts_id:4, df_id:3)
 100.00% |=====>| 1280/1280 blks 0.00s
Synchronizing...

Switching logfiles...Backing up archivelog filesArchivelog log-t0-r0-s19.arc (thread: 0, low_tsn: 36474,
next_tsn: 36508) copied
Database incremental backup succeeded
DB disconnected
RMGR backup ends
```

Backup set 5

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자를 이용한 백업/복구 예제 - Backup 정보 조회

### - BACKUPSET 정보

```
SQL> SET LINESIZE 200
SQL> COL "SIZE(KB)" FOR 99999999
SQL> COL START_TIME FOR A12
SQL> COL FINISH_TIME FOR A12
SQL> select * from V$BACKUP_SET;
```

| SET_ID | START_TIME | FINISH_TIME | START_TSN | FINISH_TSN | RESETLOGS_TSN | BASE_SET | SIZE(KB) | IS_PARTIAL | IS_INCREMENTAL | WITH_ARCHIVELOG |
|--------|------------|-------------|-----------|------------|---------------|----------|----------|------------|----------------|-----------------|
| 1      | 2019/04/05 | 2019/04/05  | 36340     | 36356      | 0             | 0        | 453588   | NO         | NO             | YES             |
| 2      | 2019/04/05 | 2019/04/05  | 36385     | 36401      | 0             | 1        | 23738    | NO         | YES            | YES             |
| 3      | 2019/04/05 | 2019/04/05  | 36415     | 36431      | 0             | 2        | 23738    | NO         | YES            | YES             |
| 4      | 2019/04/05 | 2019/04/05  | 36448     | 36464      | 0             | 1        | 23738    | NO         | YES            | YES             |
| 5      | 2019/04/05 | 2019/04/05  | 36489     | 36505      | 0             | 4        | 23738    | NO         | YES            | YES             |

### - BACKUPSET ARCHIVELOG 정보

```
SQL> SELECT SET_ID, THREAD#, MIN_LOG_TSN, MAX_LOG_TSN, MIN_LOG_SEQUENCE
, MAX_LOG_SEQUENCE, RESETLOG_TSN
FROM V$BACKUP_ARCHIVED_LOG;
```

| SET_ID | THREAD# | MIN_LOG_TSN | MAX_LOG_TSN | MIN_LOG_SEQUENCE | MAX_LOG_SEQUENCE | RESETLOG_TSN |
|--------|---------|-------------|-------------|------------------|------------------|--------------|
| 1      | 0       | 36233       | 36359       | 3                | 3                | 0            |
| 2      | 0       | 36366       | 36404       | 7                | 7                | 0            |
| 3      | 0       | 36411       | 36434       | 11               | 11               | 0            |
| 4      | 0       | 36441       | 36467       | 15               | 15               | 0            |
| 5      | 0       | 36474       | 36508       | 19               | 19               | 0            |

### - BACKUPSET ARCHIVELOG 정보

```
SQL> SELECT *
FROM V$BACKUP_SET_TABLESPACE
WHERE SET_ID=1;
```

| SET_ID | TS# |
|--------|-----|
| 1      | 0   |
| 1      | 1   |
| 1      | 3   |
| 1      | 4   |

# 복구 관리자(RMGR) 를 이용한 백업/복구

- 복구 관리자를 이용한 백업/복구 예제 - Automatic Recovery

```
[tibero@T1:/tibero]$ tbrmgr recover -o /tibero/s/backup/rmgr
=====
= Recovery Manager(RMGR) starts =
= =
= TmaxData Corporation Copyright (c) 2008-. All rights reserved. =
=====
RMGR - recovery
=====
Tibero instance terminated (ABNORMAL mode).

info file is deleted.
unlink failed.: No such file or directory
Control file #0 (/tibero/tbdata/tibero/c1.ctl) is accessible
All control files are accessible. No need to restore the backup control file.
Change core dump dir to /tibero/tibero6/bin/prof.
Listener port = 8629

Tibero 6

TmaxData Corporation Copyright (c) 2008-. All rights reserved.
Tibero instance started up (MOUNT mode).
DB connected

RMGR BEGIN RESTORE
full backup set_id: 1
last incremental backup set_id: 5
```

# 복구 관리자(RMGR) 를 이용한 백업/복구

## ● 복구 관리자를 이용한 백업/복구 예제 - Automatic Recovery

```
Applying FULL BACKUP (set_id:1, ts_id:0, df_id:0)
100.00% |=====>| 12800/12800 blks 0.20s
Synchronizing...
Applying FULL BACKUP (set_id:1, ts_id:1, df_id:1)
100.00% |=====>| 25600/25600 blks 0.61s
Synchronizing...
Applying FULL BACKUP (set_id:1, ts_id:3, df_id:2)
100.00% |=====>| 12800/12800 blks 0.00s
Synchronizing...
Applying FULL BACKUP (set_id:1, ts_id:4, df_id:3)
100.00% |=====>| 1280/1280 blks 0.00s
Synchronizing...
Applying INCREMENTAL BACKUP (set_id:4, ts_id:0, df_id:0)
100.00% |=====>| 12800/12800 blks 0.40s
Synchronizing...
Applying INCREMENTAL BACKUP (set_id:4, ts_id:1, df_id:1)
100.00% |=====>| 25600/25600 blks 0.20s
Synchronizing...
Applying INCREMENTAL BACKUP (set_id:4, ts_id:3, df_id:2)
100.00% |=====>| 12800/12800 blks 0.20s
Synchronizing...
Applying INCREMENTAL BACKUP (set_id:4, ts_id:4, df_id:3)
100.00% |=====>| 1280/1280 blks 0.00s
Synchronizing...
Applying INCREMENTAL BACKUP (set_id:5, ts_id:0, df_id:0)
100.00% |=====>| 12800/12800 blks 0.00s
Synchronizing...
Applying INCREMENTAL BACKUP (set_id:5, ts_id:1, df_id:1)
100.00% |=====>| 25600/25600 blks 0.00s
Synchronizing...
```

Backup set 1

Backup set 4

Backup set 5

# 복구 관리자(RMGR) 를 이용한 백업/복구

- 복구 관리자를 이용한 백업/복구 예제 - Automatic Recovery

```
Applying INCREMENTAL BACKUP (set_id:5, ts_id:3, df_id:2)
100.00% |=====>T 12800/12800 blks 0.00s
Synchronizing...
Applying INCREMENTAL BACKUP (set_id:5, ts_id:4, df_id:3)
100.00% |=====>T 1280/1280 blks 0.20s
Synchronizing...
```

```
Database restore succeeded
recoverSQL: ALTER DATABASE RECOVER AUTOMATIC
Database automatic recovery succeeded
DB disconnected
```

```
Tibero instance terminated (NORMAL mode).
```

```
Change core dump dir to /tibero/tibero6/bin/prof.
Listener port = 8629
```

```
Tibero 6
```

```
TmaxData Corporation Copyright (c) 2008-. All rights reserved.
Tibero instance started up (NORMAL mode).
RMGR recovery ends
```

감사합니다