

# Tibero Installation

# Contents

- 0. 교육소개
- 1. Tiberio Installation Overview
- 2. Tiberio Installation
- 3. 기동, 종료
- 4. Tiberio Directory 구조



0.

## 교육소개

| 과정                  | 내용   |
|---------------------|--|
| TIBERO Installation | <p>티베로 설치 방법을 학습합니다.</p> <ul style="list-style-type: none"><li>• 교육기간 : 1일</li><li>• 교육대상 : Tiberο 사용자</li></ul> |

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

**1.**

# **Tibero Installation Overview**

# Tibero 설치 가능 O/S

- 지원 플랫폼 및 운영체제

| H/W,S/W   | CPU                   | OS  | Binary Bits |
|-----------|-----------------------|---|-------------|
| HP        | Itanium               | HP-UX 11i v2 (11.23)<br>HP-UX 11i v3 (11.31)                      | 64Bit       |
| SUN       | SPARC                 | Solaris 10, 11  | 64Bit       |
|           | x86                   | Solaris 10  | 64Bit       |
| IBM       | PPC                   | AIX 5.3 , 6.1 , 7.1   | 64Bit       |
| GNU       | X86(Pentium4 이상)      | Red Hat Enterprise Linux 4 ~ 6<br>SUSE Linux Enterprise Server 10 | 32bits      |
|           |                       | Red Hat Enterprise Linux 4 ~ 7<br>SUSE Linux Enterprise Server 10 | 64bits      |
|           | Itanium               | Red Hat Enterprise Linux 4 ~ 7                                    | 64bits      |
|           | Power8 (Big / Little) | Red Hat Enterprise Linux 7  | 64bits      |
| Microsoft | X86, AMD64(X64)       | Windows 7<br>Windows 2000   | 32Bit       |
|           |                       | Windows 7<br>Windows 2008<br>Windows 2012                         | 64Bit       |

\*\* Windows 32bit의 경우 클라이언트만 지원한다.

# Tibero 설치 요구사항

● H/W 및 S/W 요구사항

| OS            | RAM | Swap Space | /tmp Directory Space | HDD Space<br>(Full / Client Only) |
|---------------|-----|------------|----------------------|-----------------------------------|
| LINUX         | 1GB | 2GB        | 500MB                | 2.5GB / 400M                      |
| HP-UX/itanium | 1GB | 2GB        | 500MB                | 3GB / 400M                        |
| Solaris       | 1GB | 2GB        | 500MB                | 2.5GB / 400M                      |
| AIX           | 1GB | 2GB        | 500MB                | 2.5GB / 400M                      |
| Windows       | 1GB | -          | -                    | 2GB / 400M                        |

| OS            |         | OS Version   | Compiler                                     | JDK Version          |
|---------------|---------|--|--|----------------------|
| LINUX         | x86     | Red Hat Enterprise Linux 4 kernel 2.6.9 이상   | gcc version 3.4.6 이상, C99 지원                 | JDK 1.5.17 이상        |
|               | Itanium | Red Hat Enterprise Linux 4 kernel 2.6.9 이상   | gcc version 3.4.6 이상, C99 지원                 | JDK 1.5.17 이상        |
|               | Power8  | Red Hat Enterprise Linux 7 kernel 3.10.0 이상  | gcc version 4.8.5 이상, C99 지원                 | JDK 1.5.17 이상        |
| HP-UX/itanium |         | HP-UX 11i v2 (11.23) with BUNDLE11i (== HP-UX 11i v2 (B.11.23), September 2004) 64Bit 커널 | HP aC++/ANSI C B3910B A.06.02, C99 지원        | JDK 1.5 for HP-UX 이상 |
| Solaris       |         | Solaris 10,11 64Bit 커널   | Sun C 5.8 2005/10/13, C99 지원                 | JDK 1.5.17 이상        |
| AIX           |         | AIX 5L version 5.3, AIX 6.1, AIX 7.1 64Bit 커널  | IBM XL C/C++ Enterprise Edition V7.0, C99 지원 | JDK 1.5.17 이상        |
| Windows       |         | Windows 7<br>Windows Server 2008<br>Windows Server 2012(64bit Only)                      | C99 지원 컴파일러                                  | JDK 1.5.17 이상        |

# Tibero 설치 준비사항

- Tibero를 설치하기 전에 확인하고 설정해야 할 준비사항
  - 디스크 용량 확인
  - JDK 설치
  - 운영 체제별 패키지 설치
  - 운영체제별 커널 파라미터 설정
  - 운영체제별 Shell Limits 설정
  - 호스트명과 포트 번호, 시스템 계정 확인



# Tibero 설치 준비사항

- 커널 파라미터(kernel parameter) 설정

- kernel parameter 확인 명령어

```
# sysctl -a
```

- fs.file-max : 커널이 할당할 수 있는 파일 핸들 최대값 (권장값 : 65536) / 파일(/proc/sys/fs/file-max)
  - kernel.sem : 세마포어 관련 설정. semmsl -> semmns -> semopm -> semmni 순서로 설정 파일(/proc/sys/kernel/sem)

- LINUX 커널 파라미터 설정

| 커널 파라미터                              | 값                                | 파일                                     |
|--------------------------------------|----------------------------------|--|
| semmsl<br>semmns<br>semopm<br>semmni | 10000<br>32000<br>10000<br>10000 | /proc/sys/kernel/sem                   |
| shmall                               | ceil(shmmax/PAGE_SIZE)           | /proc/sys/kernel/shmall                |
| shmmax                               | 물리적인 메모리의 절반 (byte)              | /proc/sys/kernel/shmmax                |
| shmmni                               | 4096                             | /proc/sys/kernel/shmmni                |
| file-max                             | 6815744                          | /proc/sys/fs/file-max                  |
| ip_local_port_range                  | 1024 ~ 65000                     | /proc/sys/net/ipv4/ip_local_port_range |

# Tibero 설치 준비사항

## ● 커널 파라미터(kernel parameter) 수정

- root 로 접속하여 아래 제시한 권장 값을 기준으로 각 사이트 운영 환경에 맞추어 설정.
- kernel parameter 동적 변경

| 확인하기                                   | 변경하기   |
|--|--|
| # sysctl -a   grep sem                 | ➡ # sysctl -w kernel.sem="10000 32000 10000 10000"       |
| # sysctl -a   grep shmall              | ➡ # sysctl -w kernel.shmall= 655360                      |
| # sysctl -a   grep shmmax              | ➡ # sysctl -w kernel.shmmax= 2684354560                  |
| # sysctl -a   grep shmmni              | ➡ # sysctl -w kernel.shmmni=4096                         |
| # sysctl -a   grep file-max            | ➡ # sysctl -w fs.file-max=6815744                        |
| # sysctl -a   grep ip_local_port_range | ➡ # sysctl -w net.ipv4.ip_local_port_range ="1024 65000" |

### ➤ kernel parameter 정적 변경

- 시스템 환경파일을 vi 에디터

```
# vi /etc/sysctl.conf
```

- kernel parameter 항목을 sysctl.conf 파일에 추가.

```
fs.file-max = 65536  
kernel.sem = 10000 32000 10000 10000  
kernel.shmmax = 2684354560
```

- sysctl.conf에 추가한 후 **sysctl -p** 명령으로 동적으로 적용 가능

# Tibero 설치 준비사항

## ● Shell Limits 설정

- 프로세스의 리소스 사용 제한
- 소프트 한도 : 새로운 프로세스가 만들어졌을 때 디폴트로 적용
- 하드 한도 : OS 유저가 소프트 한도 이상으로 큰 값을 적용할 수 있으나, 최대 크기는 하드 한도 까지 임

### ➤ 설정 파일

```
/etc/security/limits.conf
```

### ➤ 설정 값

| 파라미터          | Soft Limit | Hard Limit |
|---------------|------------|------------|
| <b>nofile</b> | 1024       | 65536      |
| <b>nproc</b>  | 2047       | 16384      |

# Tibero 설치 준비사항

- 호스트명과 포트 번호, 시스템 계정 확인

- 라이선스를 요청할 때 필요한 호스트명을 확인. /etc/hosts 파일을 확인하거나 콘솔 화면에서 명령어 `uname -n` 를 입력하여 확인.

- 포트 번호

- Tibero가 기동할 때 부여될 포트 번호를 확인한다. (기본값 : 8629)

| 종류                   | 설명  | 사용시점         |
|----------------------|---|--------------|
| 서비스 포트(Service port) | Tibero가 사용자의 요청을 처리하고 응답할 포트 번호를 확인한다.<br>(기본값 : 8629)  | All the time |
| 스페셜 포트(Special port) | Session Full 과 같은 상황에서 응급조치를 하기 위해 사용할 포트를 확인한다. tip 파일에 <b>_LSNR_SPECIAL_PORT</b> 를 이용한 임의 설정도 가능하다.<br>(기본값 : 서비스 포트 + 1) | All the time |
| SSL 포트(SSL port)     | 보안 통신을 위해 사용할 포트를 확인한다. tip 파일에 <b>_LSNR_SSL_PORT</b> 를 이용한 임의 설정도 가능하다.<br>(기본값 : 서비스 포트 + 2)                              | 보안통신 사용시     |
| CM 포트(CM port)       | CM이 사용할 포트를 확인한다. tip 파일에 <b>CM_PORT</b> 를 이용한 임의 설정도 가능하다.<br>(기본값 : 서비스 포트 + 3)   | TAC 사용시      |

- 시스템 계정 확인 : Tibero 설치 및 운영을 위한 계정을 확인.

# Tibero 설치 준비사항

- 그룹 및 사용자 생성

- 그룹 추가 (권장 값 : dba)

```
# groupadd dba
```

- 사용자 추가 (권장 값 : tibero)

```
# useradd -g dba tibero
```

- 사용자 환경변수 설정

- 사용자 계정 홈 디렉토리있는 환경파일에 다음과 같은 내용을 추가함(bash shell일 경우 .bash\_profile)

```
### JAVA ENV ###  
export JAVA_HOME=/usr/java6  
export JDK_HOME=$JAVA_HOME  
export PATH=$JAVA_HOME/bin:/usr/local/bin:$PATH
```

# Tibero 설치 준비사항

## ● TIBERO 환경 변수

- TB\_HOME : Tibero 소프트웨어가 설치된 디렉토리
- TB\_SID : Tibero System Identify 서비스 명
- TB\_PROF\_DIR : core 파일이 발생할 경우 파일 생성 위치 지정
- PATH : Tibero 서버를 사용하기 위한 실행파일 경로 설정
- LD\_LIBRARY\_PATH : Tibero 서버 사용시 필요한 공유 라이브러리 패스 설정

# Tibero 설치 준비사항

● TIBERO 파라미터

| 파라미터              | 설명  |
|-------------------|---|
| CONTROL_FILES     | 컨트롤 파일이 존재하는 위치로서 절대 경로로 설정한다.            |
| DB_NAME           | 데이터베이스 이름을 설정한다. (기본값: tibero)            |
| MAX_SESSION_COUNT | 데이터베이스에 접속 가능한 최대 세션 수를 설정한다. (기본값: 20)   |
| MEMORY_TARGET     | 데이터베이스가 사용할 수 있는 메모리의 총 크기를 설정한다.         |
| TOTAL_SHM_SIZE    | 데이터베이스의 인스턴스 내에서 사용할 전체 공유 메모리의 크기를 설정한다. |
| LISTENER_PORT     | 리스너가 사용할 포트 번호를 설정한다. (기본값: 8629)         |

# Tibero 설치 준비사항

## ● TIBERO 데이터베이스 생성 항목

| 항목                            | 설명   |
|-------------------------------|--|
| <b>SYS password</b>           | Tibero 시스템 사용자의 패스워드                       |
| <b>Character Set</b>          | 데이터베이스의 문자 집합(기본값: MSWIN949)               |
| <b>National Character Set</b> | 데이터베이스의 내셔널 문자 집합(기본값: UTF16)              |
| <b>MAXDATAFILES</b>           | 데이터베이스에서 사용할 데이터 파일의 최대값을 설정한다. (기본값: 100) |
| <b>MAXLOGGROUPS</b>           | 로그 그룹의 최대값을 설정한다. (기본값: 255)               |
| <b>MAXLOGMEMBERS</b>          | 로그 그룹 내의 로그 파일의 최대값을 설정한다. (기본값: 8)        |
| <b>ARCHIVE LOG mode</b>       | 로그 아카이브 모드를 설정한다. (기본값: NO)                |
| <b>LOG Group</b>              | LOG GROUP을 설정한다.                           |
| <b>SYSTEM Tablespace</b>      | 시스템 테이블 스페이스의 데이터 파일을 설정한다.                |
| <b>TABLESPACE</b>             | 테이블 스페이스를 설정한다.                            |



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

2.

## Tibero Installation

# Tibero 설치 과정

1. 설치 파일 준비 ( Tibero 바이너리, 라이선스 파일 등 )
2. 환경설정 파일에 환경변수 설정
3. Tibero 바이너리 압축해제, 라이선스 파일 복사
4. 파라미터 파일 생성용 shell 실행
5. Tibero 인스턴스 기동( nomount 모드 )
6. Database 생성
7. Tibero 인스턴스 기동( normal 모드 )
8. Sytem object 생성용 shell 실행

# Tibero 설치

- 사용자 환경변수 설정

- 사용자 계정 홈 디렉토리있는 환경파일 내용 추가(bash shell일 경우 .bash\_profile)

```
##### TIBERO ENV #####
export TB_HOME=/tibero/tibero6
export TB_SID=tibero
export TB_PROF_DIR=$TB_HOME/bin/prof
export PATH=.:$TB_HOME/bin:$TB_HOME/client/bin:~/tbinary/monitor:$PATH
export LD_LIBRARY_PATH=$TB_HOME/lib:$TB_HOME/client/lib:$LD_LIBRARY_PATH
export SHLIB_PATH=$LD_LIBRARY_PATH:$SHLIB_PATH
export LIBPATH=$LD_LIBRARY_PATH:$LIBPATH

##### TIBERO alias #####
alias tbhome='cd $TB_HOME'
alias tbbin='cd $TB_HOME/bin'
alias tblog='cd $TB_HOME/instance/$TB_SID/log'
alias tbcfg='cd $TB_HOME/config'
alias tbcfgv='vi $TB_HOME/config/$TB_SID.tip'
alias tbcli='cd ${TB_HOME}/client/config'
alias tbcliv='vi ${TB_HOME}/client/config/tbdsn.tbr'
alias tbcliv='vi ${TB_HOME}/client/config/tbnet_alias.tbr'
alias tbdata='cd $TB_HOME/tbdata'
alias tbi='cd ~/tbinary'
alias clean='tbdwn clean'
alias dba='tbsql sys/tibero'
alias tm='cd ~/tbinary/monitor;monitor;cd -'
```

# Tibero 설치 준비사항

- 사용자 환경변수 설정(계속)

- 환경변수 적용

```
$ source ~/.bash_profile
```

- 환경변수 확인

- .bash\_profile 의 환경변수가 적용되었는지 확인한다.

```
$ echo $TB_SID  
tibero  
  
$ echo $TB_HOME  
/tibero/tibero6
```

- 티베로 바이너리, 라이선스 파일 준비

- 티베로바이너리파일, 라이선스 파일을 준비하여 /tibero 디렉토리에 위치시킨다.

```
$ ls -l  
  
-rw-r--r--  1 tibero dba      471 Oct 10 11:12 license.xml  
-rwxr-x---  1 tibero dba 386220636 May 15 17:24 tibero6-bin-FS06-linux64-151366-opt-tested.tar.gz
```

# Tibero 설치

## ● Tibero 바이너리 설치

- 티베로에 접속하여 티베로 압축 파일 해제 .

```
$ cd /tibero
$ tar -xvzf tibero6-bin-FS06-linux64-151366-opt-tested.tar.gz
tibero6/bin/nbboot
tibero6/bin/alterdd.sh
.....
```

```
tibero6      → $TB_HOME
| --- bin
| --- client
| --- config
| --- scripts
```

## ● 라이선스 적용

- Tibero를 설치한 hostname에 맞는 license.xml 파일을 발급
- \$TB\_HOME/license 디렉토리 생성 후 라이선스 파일을 이 디렉토리에 복사

## ● 초기 환경파일 생성

```
$ cd $TB_HOME/config
$ ./gen_tip.sh
```

- \$TB\_SID.tip 파일 생성 : Tibero 파라미터 파일
- tbdsn.tbr 파일 생성 : Tibero Client 접속 설정 파일

# Tibero 설치 (계속)

- Tibero 파라미터 파일(\$TB\_HOME/config/\$TB\_SID.tip) 수정

```
DB_NAME=tibero
LISTENER_PORT=8629
CONTROL_FILES="/tibero/tbdata/tibero/c1.ctl","/tibero/tbdata/tibero/c2.ctl"
DB_CREATE_FILE_DEST=/tibero/tbdata/tibero
LOG_ARCHIVE_DEST=/tibero/tbdata/tibero/arch
MAX_SESSION_COUNT=20
TOTAL_SHM_SIZE=600M
MEMORY_TARGET=1G

.... 중략 ....
```

- Network 설정 파일(\$TB\_HOME/client/config/tbdsn.tbr) 수정 가능

- Client가 Tibero에 접속 시 필요한 통신 환경을 설정하는 파일

```
tibero=(
    (INSTANCE=(HOST=localhost)          # SID name
      (PORT=8629)                        # 접속할 Tibero IP
      (DB_NAME=tibero)                  # 접속할 Tibero Port
    )                                   # 접속할 Tibero Database 명
)
```

# Tibero 설치 (계속)

- Tibero 데이터베이스 생성

- Tibero를 nomount 모드로 부팅

```
$ tbboot nomount  
listener port = 8629  
change core dump dir to /tibero/tibero6/bin/prof  
  
Tibero 6  
  
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.  
  
Tibero instance started suspended at NOMOUNT mode.
```

- System 유저로 접속

```
$ tbsql sys/tibero  
  
tbSQL 6  
  
Copyright (c) 2008, 2009, 2011 Tibero Corporation. All rights reserved.  
  
Connected to Tibero.  
  
SQL>
```

# Tibero 설치 (계속)

## ➤ Database 생성 스크립트

```
SQL> CREATE DATABASE
      USER sys IDENTIFIED BY tibero
      MAXDATAFILES 256
      CHARACTER SET MSWIN949      -- UTF8, EUCKR, ASCII ...
      NATIONAL CHARACTER SET UTF16
      LOGFILE
        GROUP 0 'log01.log' SIZE 50M,
        GROUP 1 'log11.log' SIZE 50M,
        GROUP 2 'log21.log' SIZE 50M
      MAXLOGFILES 100
      MAXLOGMEMBERS 2
      NOARCHIVELOG
        DATAFILE 'system001.dtf' SIZE 100M
        AUTOEXTEND ON NEXT 64M MAXSIZE 3G
      DEFAULT TEMPORARY TABLESPACE TEMP
        TEMPFILE 'temp001.dtf' SIZE 100M
        AUTOEXTEND ON NEXT 64M MAXSIZE 3G
        EXTENT MANAGEMENT LOCAL AUTOALLOCATE
      UNDO TABLESPACE UNDO
        DATAFILE 'undo001.dtf' SIZE 200M
        AUTOEXTEND ON NEXT 64M MAXSIZE 3G
        EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128k
      DEFAULT TABLESPACE USR
        DATAFILE 'usr001.dtf'   SIZE 50m
        AUTOEXTEND ON NEXT 64m MAXSIZE 3G
      SYSSUB
        DATAFILE 'syssub001.dtf' SIZE 50m
        AUTOEXTEND ON NEXT 64M MAXSIZE 3G ;
```



# Tibero 설치 (계속)

## ➤ Tibero 재기동

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

## ➤ Data Dictionary 및 system 패키지 생성

```
$ cd $TB_HOME/scripts  
$ sh system.sh  
Creating the role DBA...  
Creating system users & roles...  
Creating virtual tables(1)...  
Creating virtual tables(2)...  
Granting public access to _VT_DUAL...  
Creating the system generated sequences...  
Creating internal dynamic performance views...  
Creating system packages:  
Running /home/tibero6/tibero6/scripts/pkg/pkg_stan
```

### ❖ 주의사항

~ sh system.sh 실행시 2회의 패스워드 입력을 받으며,  
다음과 같이 입력해야 함.

✓ 첫번째 sys 유저의 패스워드 입력 →  
**tibero**

✓ 두번째 syscat 패스워드 입력 →  
**syscat**

~ 다른 항목은 모두 Y 입력후 엔터 키를 입력함.

# Tibero 설치 (계속)

➤ system.sh 수행 옵션

| 옵션           | 설명  |
|--------------|---|
| -p1 password | sys 계정의 패스워드를 입력한다. (기본값: tibero)   |
| -p2 password | syscat 계정의 패스워드를 입력한다. (기본값: syscat)  |
| -a1          | 시스템 유저 생성 및 권한부여 여부를 설정한다. 최초 설치일 경우 반드시 'Y'로 설정한다.<br>Y: 시스템 유저 생성 및 권한을 부여한다.<br>N: 데이터를 보존한다.                                |
| -a2          | 데이터베이스 사용자의 패스워드 관리 정책을 지정할 수 있는 프로파일을 위한 테이블의 생성 여부를 설정한다. 최초 설치일 경우 반드시 'Y'로 설정한다.<br>Y: 기존 테이블을 삭제한 후 생성한다.<br>N: 데이터를 보존한다. |
| -a3          | 데이터베이스 객체의 통계 정보 수집을 위한 Job 스케줄링 등록 여부를 설정한다. 최초 설치일 경우 'Y'로 설정하는 것을 권장한다.<br>Y: 통계정보 수집을 위한 Job을 삭제한 후 등록한다.<br>N: 등록하지 않는다.   |

# Tibero 설치 (계속)

➤ system.sh 수행 옵션

| 옵션      | 설명  |
|---------|---|
| -a4     | 통계정보를 주기적으로 자동 수집하기 위한 TPR 관련 테이블의 생성여부를 설정한다. 최초 설치일 경우 반드시 'Y'로 설정한다.<br>Y: 기존 테이블을 삭제한 후 생성한다.<br>N: 데이터를 보존한다.                                  |
| pkgonly | PSM built-in package 스크립트들만 수행한다.   |
| -sod    | 보안을 위하여 기존 하나의 관리자 권한을 아래와 같이 3개의 관리자로 나뉘어 설치할 것인지에 대한 여부를 설정한다. (기본값: N, Windows 계열에서는 미지원)<br>SYSSEC: 보안 관리자<br>SYSAUD: 감사 관리자<br>SYSADM: 시스템 관리자 |
| error   | system.sh 수행 중 특정 스크립트에서 에러가 발생할 경우 에러 내용을 터미널에 출력한다. (Windows 계열에서는 미지원)   |

# Tibero 설치 (계속)

- 설치 에러 해결 방법

- 사용하는 port 를 다른 곳에서 사용할 경우

**listener port = 8629**

**Error: timeout while trying to open port 6999**

**Tibero instance startup failed!**

- 다른곳에서 해당 포트번호를 사용을 하고 있는지 확인
- 명령어> netstat -an | grep 8629
- 2개 파일에서 포트 번호를 변경하여 처리함.
  - \$TB\_HOME/config/tibero.tip
  - \$TB\_HOME/client/config/tbnet\_alias.tbr

- Shared Memory 할당 못 받는 현상

**Cannot attach to shared memory segment: SHM\_KEY = 0x1a71a8f3, size = 335544320, errno = 29Tibero instance startup failed!**

- 해당 에러는 티베로가 종료된 상태임에도 공유 메모리 해제를 제대로 못해주면서 발생한것으로써 ipcs 로 shared memory 에 존재하는지 확인하여야 한다.
- ipcrm 명령어로 해당 자원을 해제하도록 한다.
  - ex) ipcrm -m 32768
  - ipcrm -s 131072

# Tibero 제거

- Tibero 제거

- Tibero프로세스 종료

```
$ tbdown 또는 tbdown immediate
```

- \$TB\_HOME 에 해당하는 디렉토리를 포함한 하위 모든 디렉토리를 OS 명령으로 삭제

```
$ rm -rf $TB_HOME
```

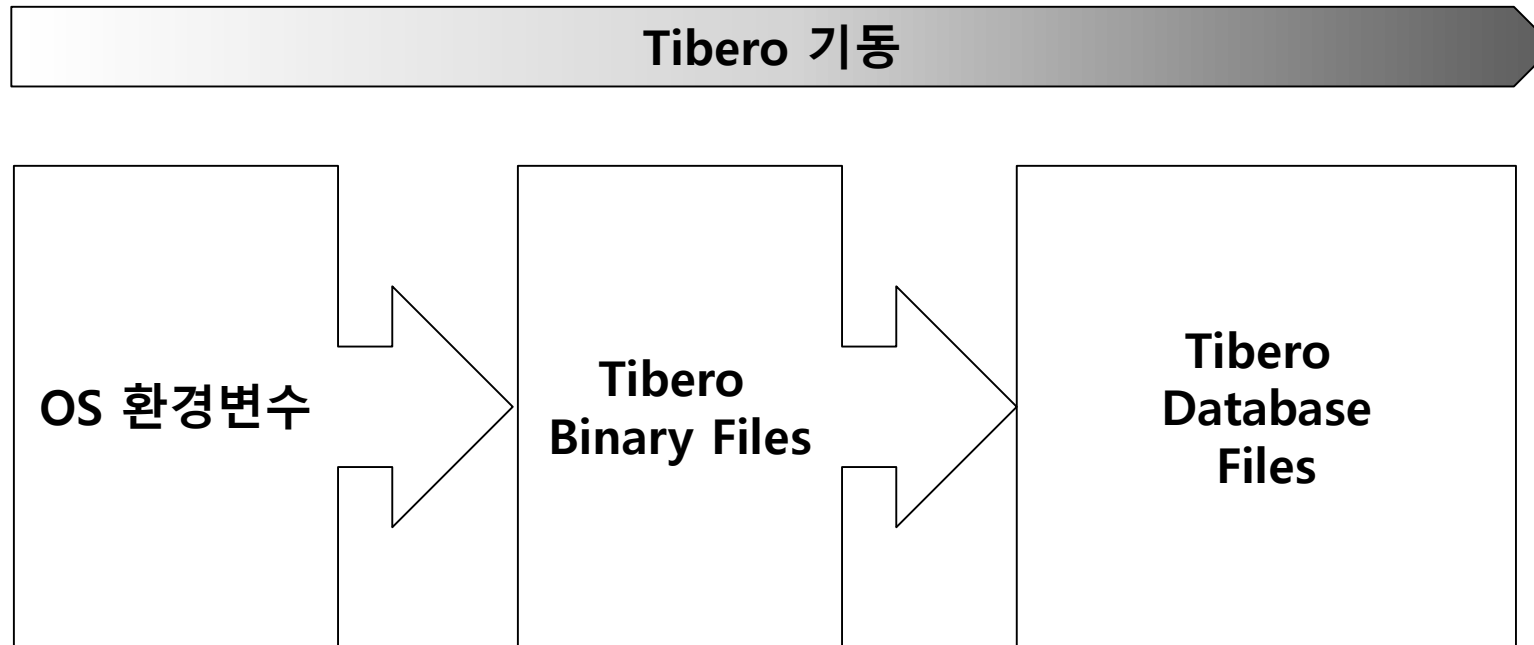
A blue abstract graphic with a textured, brush-stroke-like appearance, featuring shades of blue and purple.

3.

Tibero 기동, 종료

# Tibero 기동 환경

Tibero 기동을 위해서는 OS 커널파라미터 및 환경변수 그리고 Tibero 바이너리, 데이터베이스 파일이 정상적으로 설정 및 설치가 되어 있어야 함



# Tibero 기동 환경

Tibero 는 OS 환경변수를 참조하여 구동 되므로, 적절한 값이 설정되어 있어야 한다

| 환경변수            | 설명  |
|-----------------|---|
| TB_HOME         | Tibero가 설치된 디렉터리이다.<br><b>예: /tibero/tibero6</b>  |
| TB_SID          | Tibero 인스턴스 식별자 이며, 파라미터 파일의 이름, 각종 로그 생성 경로 등에 영향을 준다<br><b>예: tibero</b>  |
| LD_LIBRARY_PATH | Tibero를 사용할 때 필요한 공유 라이브러리가 위치한 경로이다.<br>필요한 라이브러리는 모두 \$TB_HOME/lib와 \$TB_HOME/client/lib 안에 있고, OS별로 환경변수가 다르게 지정된다.<br><br>SunOS, Linux : LD_LIBRARY_PATH<br>HP-UX : SHLIB_PATH<br>AIX : LIBPATH |
| PATH            | Tibero 의 실행파일 디렉터리 경로이다. OS 기본 \$PATH 에 \$TB_HOME/bin 와 \$TB_HOME/client/bin 을 추가하여 설정한다<br><b>예: /tibero/tibero6/bin , /tibero/tibero6/client/bin</b>  |



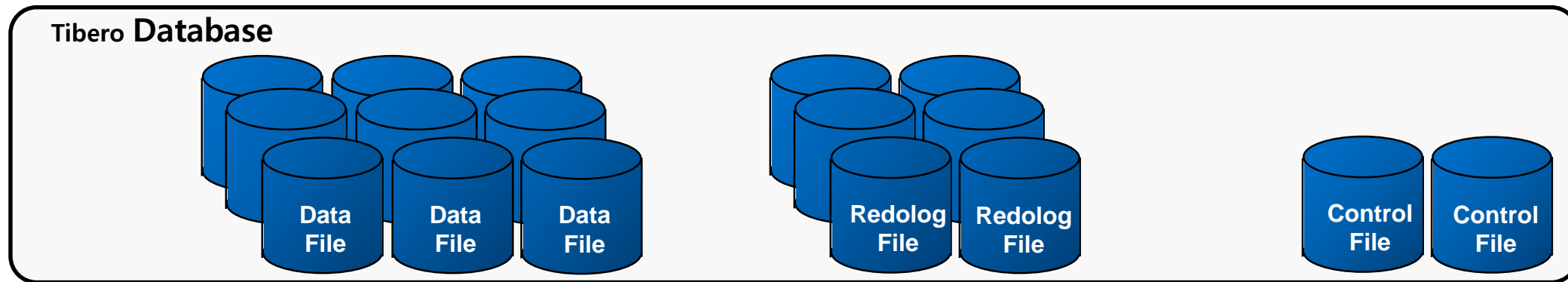
# Tibero 기동 환경

Tibero 서버는 다음과 같은 디렉터리로 구성되며, 구동을 위해서는 Tibero 바이너리 파일이 정상적으로 설치되어 있어야 한다

```
[tibero@T1:/tibero/examples/tibero6]$ ls -l
total 12
drwxr-xr-x  4 tibero dba 4096 Jan  4 15:54 bin
drwxr-xr-x 10 tibero dba  105 Jan  4 15:37 client
drwxr-xr-x  2 tibero dba 4096 Feb 19 15:26 config
drwxr-xr-x 11 tibero dba  148 Jan 15 10:30 instance
drwxr-xr-x  3 tibero dba  249 Jan  4 15:37 lib
drwxr-xr-x  3 tibero dba   45 Jan  4 15:54 license
drwxr-xr-x  3 tibero dba   22 Jan  4 15:37 nls
drwxr-xr-x  3 tibero dba 4096 Jan 15 10:56 scripts
```

# Tibero 기동 환경

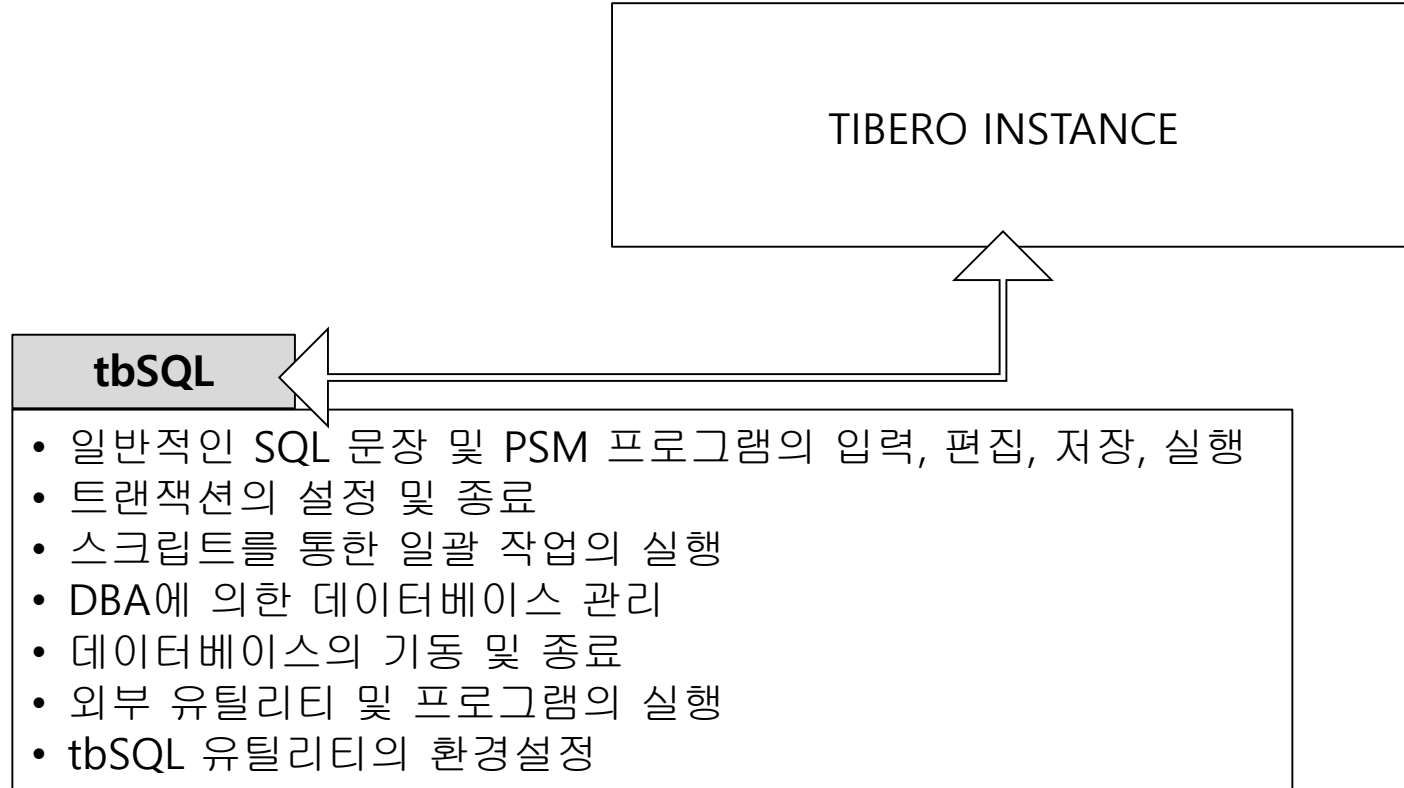
Tibero 서버 기동을 위해서는 데이터베이스 파일이 정상적으로 생성되어 있어야 한다



- Datafiles
  - Tables & Indexes 등과 같은 데이터 저장 객체를 물리적으로 저장하고 있는 파일.
- Redo Log Files
  - 복구를 위하여 Database에서 변경된 모든 것을 기록 하는 파일.
- Control Files
  - Database의 물리적 구조와 상태를 기록 하는 파일.

# tbSQL 유틸리티

Tibero 관리를 위해서 사용하는 기본 도구로서 tbSQL 를 제공하고 있다. tbSQL은 TEXT 모드의 대화형 SQL 명령어 처리를 지원하는 유틸리티이다. SQL 질의, 데이터 정의어, 트랜잭션과 관련된 SQL 문장 등을 실행할 수 있다.



# Tibero 기동 종료

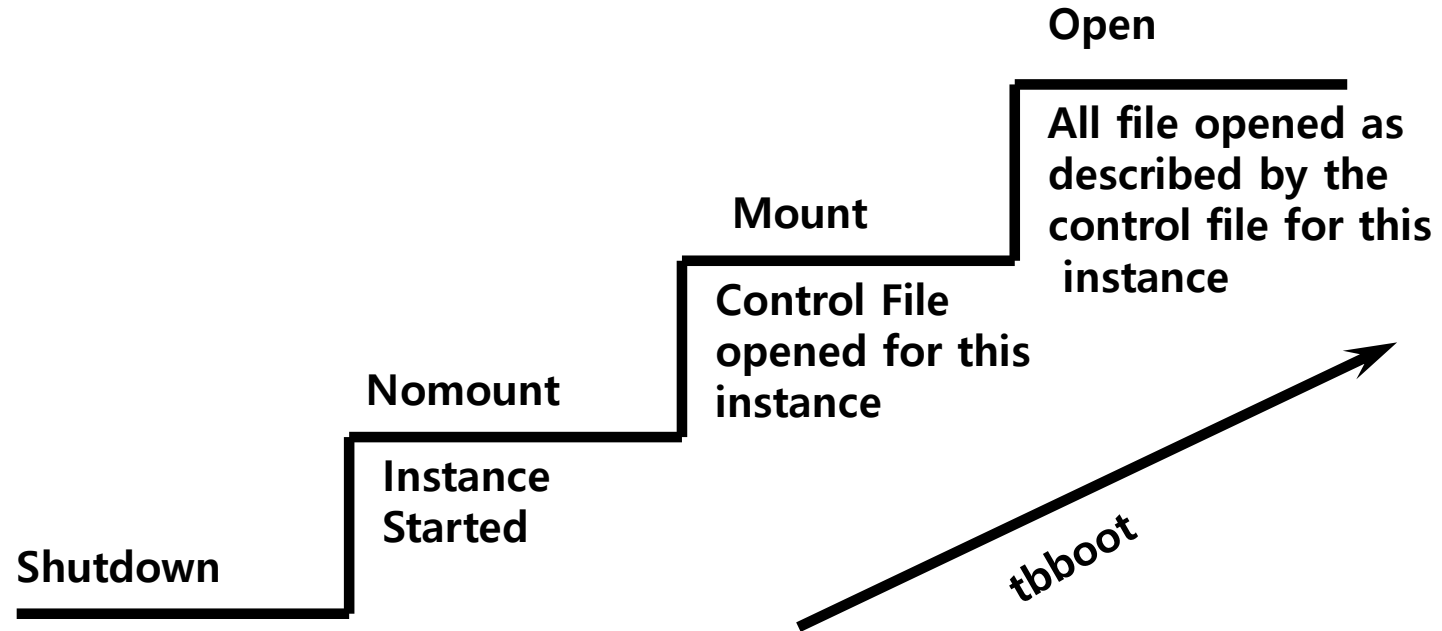
## tbboot

- tbboot는 Tibero가 설치된 머신에서 실행해야 한다.
- tbboot를 실행하기 전에 반드시 환경변수가 제대로 설정되어 있어야 한다.
- 이 명령어와 관련된 환경변수는 \$TB\_HOME과 \$TB\_SID이다.
- tbboot는 실행 파일을 실행할 수 있는 권한이 있는 사용자라면 어느 누구든 Tibero를 기동할 수 있으므로, Tibero를 설치한 사용자만 Tibero의 실행 파일에 접근할 수 있도록 권한을 설정할 것을 권장한다.

| 옵션 | 설명   |
|----|--|
|    | 옵션이 없는 경우 Tibero를 부트 모드(bootmode) 중 NORMAL로 기동하는 옵션이다.               |
| -h | tbboot를 사용하기 위한 간단한 도움말을 보여주는 옵션이다.                                  |
| -v | Tibero의 버전 정보를 보여주는 옵션이다.  |
| -C | Tibero에서 사용 가능한 character set 정보와 nls_date_lang 정보를 보여주는 옵션이다.       |
| -c | Tibero가 replication mode로 설정되어 있을 경우 replication mode를 사용하지 않는 옵션이다. |
| -t | Tibero 서버를 기동할 수 있는 옵션이다. 이 옵션은 생략이 가능하다.                            |

# Tibero 기동 종료

## Tibero Instance 기동



# Tibero 기동 종료

## Tibero Instance 종료

```
tbdown
tbdown -h
tbdown [ normal | post_tx | immediate | abort | switchover | abnormal |
        NORMAL | POST_TX | IMMEDIATE | ABORT | SWITCHOVER | ABNORMAL ]
```

| 옵션        | 설명  |       |    |        |                              |         |   |           |   |       |                             |
|-----------|---|-------|----|--------|------------------------------|---------|---|-----------|---|-------|-----------------------------|
|           | 옵션이 없는 경우 Tibero를 정상 모드로 종료하는 옵션.   |       |    |        |                              |         |   |           |   |       |                             |
| -h        | tbdown 을 사용하기 위한 간단한 도움말을 보여주는 옵션.  |       |    |        |                              |         |   |           |   |       |                             |
| -t        | Tibero 서버를 종료할 수 있는 옵션이다. 이 옵션은 생략이 가능. <table><tr><th>다운 모드</th><th>설명</th></tr><tr><td>NORMAL</td><td>모든 세션이 끝날 때까지 기다린 후 종료하는 모드.</td></tr><tr><td>POST_TX</td><td>모든 트랜잭션이 끝날 때까지 기다린 후 Tibero 를 종료하는 모드.</td></tr><tr><td>IMMEDIATE</td><td>현재 수행 중인 모든 작업을 강제로 중단시키며, 진행 중인 트랜잭션을 롤백하고 Tibero 를 종료하는 모드.</td></tr><tr><td>ABORT</td><td>Tibero 의 프로세스를 강제로 종료하는 모드.</td></tr></table> | 다운 모드 | 설명 | NORMAL | 모든 세션이 끝날 때까지 기다린 후 종료하는 모드. | POST_TX | 모든 트랜잭션이 끝날 때까지 기다린 후 Tibero 를 종료하는 모드. | IMMEDIATE | 현재 수행 중인 모든 작업을 강제로 중단시키며, 진행 중인 트랜잭션을 롤백하고 Tibero 를 종료하는 모드. | ABORT | Tibero 의 프로세스를 강제로 종료하는 모드. |
| 다운 모드     | 설명  |       |    |        |                              |         |   |           |   |       |                             |
| NORMAL    | 모든 세션이 끝날 때까지 기다린 후 종료하는 모드.  |       |    |        |                              |         |   |           |   |       |                             |
| POST_TX   | 모든 트랜잭션이 끝날 때까지 기다린 후 Tibero 를 종료하는 모드.   |       |    |        |                              |         |   |           |   |       |                             |
| IMMEDIATE | 현재 수행 중인 모든 작업을 강제로 중단시키며, 진행 중인 트랜잭션을 롤백하고 Tibero 를 종료하는 모드.   |       |    |        |                              |         |   |           |   |       |                             |
| ABORT     | Tibero 의 프로세스를 강제로 종료하는 모드.   |       |    |        |                              |         |   |           |   |       |                             |

# Tibero 기동 종료

## Tibero Instance 종료 과정

- **Database Close**
  - TSM에 있는 모든 Database Data와 Recovery Data를 Data File과 Redo Log File에 각각 기록.
- **Dismounting Database**
  - Control File을 Close.
- **Instance Shutdown**
  - TSM에 있는 Memory가 제거되고, Process가 종료.
  - Abnormal Shutdown이 되었을 경우 TSM에 Memory가 상주해 있거나 Process가 종료되지 않을 수가 있다. 따라서 남아 있는 이전 Instance를 제거하거나, tbdwn abort 명령어를 이용하여 새로운 Instance를 강제로 종료하고, tbdwn clean.

# Tibero 기동 종료

## ● tbdown abort

- Tibero의 프로세스를 강제로 종료하는 모드.
- ABORT는 Tibero에 SYS 사용자로 접속한 다음 Tibero의 MTHR 프로세스가 모든 프로세스를 OS의 강제 종료 시그널을 전달하여 강제로 종료시키는 모드. 따라서 이 모드는 비상 시에 사용하며. 다음 번에 TiberoRDBMS를 기동할 때 파손 복구 과정이 필요.
- ABORT는 Tibero가 강제로 종료시키므로 사용하던 시스템 리소스를 해제할 기회 미존재. 따라서 서버가 종료된 다음에도 공유 메모리(Shared Memory), 세마포어(Semaphore) 등의 시스템 리소스가 남아있을 수 있으며, 로그 파일이나 데이터 파일도 동일.
- tbdown abort 이후에 Tibero 재 기동 시 파손복구에 많은 시간이 소요 가능성 존재.
- ABORT 는 다음과 같은 상황에서만 사용하도록 권장.
  - ✓ Tibero 의 내부 에러로 인한 정상적인 종료가 불가능한 경우.  
(예를 들어, 모든 세션이 사용 중인 상태에서 클라이언트의 접속이 증가하는 경우.)
  - ✓ H/W에 문제가 발생하여 Tibero 를 즉시 종료해야 하는 경우.
  - ✓ 해킹 등의 비상 상태가 발생하여 Tibero 를 즉시 종료해야 하는 경우.



# Tibero 기동 종료

- **tbdown clean**

| 옵션           | 설명  |
|--------------|---|
| <b>clean</b> | Tibero 서버가 비정상 종료된 상태에서 운영중에 사용하였던 공유 메모리나 세마포어 자원들을 해제하는 옵션. Tibero 서버가 운영 중일 때는 사용 할 수 없는 옵션. |

- **kill 명령으로 종료한 경우, tbdown clean 를 실행.**

- 만약 Tibero 서버가 Kill 과 같은 시스템 내부 명령어에 의해서 비정상적으로 종료된 경우, 운영 중에 사용하였던 공유 메모리나 세마포어 자원들이 해제가 안 될 수 있다. 이런 경우 Tibero 가 정상적으로 재 기동 하지않게 되므로, tbdown clean 으로 기존 자원을 해제시킨 후 tbboot 를 실행.

## **BOOT WITH AUTO DOWN CLEAN 파라미터**

비정상적으로 종료되더라도 초기화 파라미터 BOOT\_WITH\_AUTO\_DOWN\_CLEAN를 Y로 설정하면 자동으로 이전 운영 중에 사용하였던 자원을 해제시켜 부팅 가능.

관리자가 서버의 비정상 종료 상황을 제대로 인지하지 못하고 서버 운영을 할 수 있으며, 기존 자원이 나 프로세스가 제대로 정리가 안 되는 예외적이 상황이 발생하여 충돌이 날 수 있으므로 BOOT\_WITH\_AUTO\_DOWN\_CLEAN 옵션을 켜는 것을 비 권장.

# Tibero 기동 종료

- **tbdown POST\_TX**

- 모든 트랜잭션이 끝날 때까지 기다리고 나서 Tibero를 종료하는 모드.
- POST\_TX는 Tibero에 SYS 사용자로 접속한 다음 모든 트랜잭션이 끝날 때까지 기다림.  
그 다음 Tibero RDBMS를 종료.
- tbdown 실행이 시작되면 더는 데이터베이스에 접속할 수 없고, 이미 열려 있던 세션에서도 새로운 트랜잭션을 시작할 수 없음. 다만, 현재 수행 중인 트랜잭션은 Commit 혹은 Rollback 할 때까지 제한 없이 수행할 수 있으며, Commit이나 Rollback을 하는 순간 자동으로 데이터베이스 접속을 종료.
- tbdown 실행이 시작되면 데이터베이스에 접속한 클라이언트에게 서버 종료를 알리는 메시지를 보내지 않음
- tbSQL 유틸리티 등에서는 클라이언트가 서버 종료를 즉시 알지 못하고, 그 다음 명령을 실행할 때 비로소 Tibero RDBMS가 종료되었음을 알리도록 됨

# Tibero 기동 종료

- **tbdown POST\_TX**

- 사용 예제

```
SQL> CREATE TABLE T1 (COL1 NUMBER);
```

Table 'T1' created.

```
SQL> INSERT INTO T1 VALUES(10);
```

1 row inserted.

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

COL1

-----

10

1 row selected.

...이 시점에서 **tbdown POST\_TX** 명령을 실행한다.

...**tbdown** 명령은 트랜잭션이 끝나기를 기다린다.

```
SQL> COMMIT;
```

Commit succeeded.

...이 시점에서 실제로 서버가 종료되고 **tbdown** 실행이 끝난다.

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

TBR-2069: I/O read error

...서버가 종료되었으므로 **tbSQL** 유틸리티의 프롬프트에서는 TBR-2069 에러가 발생된다.

# Tibero 기동 종료

- **tbdownd IMMEDIATE**

- 현재 수행 중인 모든 작업을 강제로 중단시키며, 진행 중인 모든 트랜잭션을 롤백하고 Tibero를 종료하는 모드.
- IMMEDIATE는 Tibero에 SYS 사용자로 접속한 다음 현재 수행 중인 모든 작업을 강제로 종료하고, 진행 중이던 모든 트랜잭션을 롤백한 후, Tibero를 종료. 클라이언트에서 Tibero 종료를 알지 못하는 것은 POST\_TX 모드와 동일.
- 트랜잭션이 오래 걸리는 작업 중에 있었다면, 이를 모두 롤백하기 위해서 다소 시간이 걸림.
- 사용 예

```
$ tbdownd immediate
```

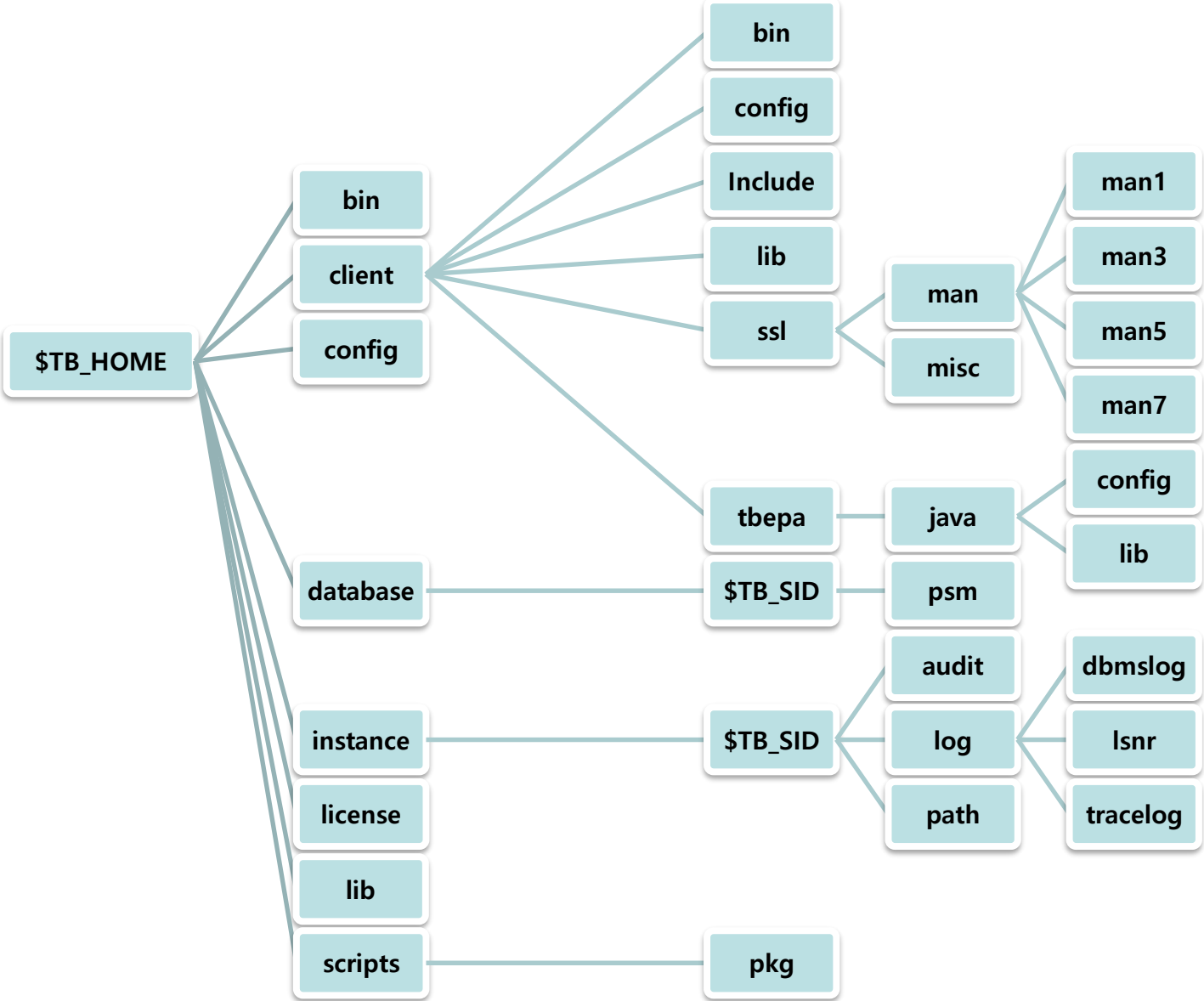
```
Tibero instance terminated (IMMEDIATE mode).
```

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

4.

## Tibero Directory 구조

# Tibero 서버 Directory 구조



# Tibero 서버 Directory 구조

- Directory 구성

| 구분         | Data Type   |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
|------------|---|------|----|-------|---|------|--|----------|---|----------|-------------------------------------|----------|--------------------------------------|------|---|
| bin        | <p>Tibero의 실행 파일과 서버 관리를 위한 유틸리티가 위치한 디렉터리이다. 이 디렉터리에 속한 파일 중에서 tbsvr과 tlistener는 Tibero를 구성하는 실행 파일이며, tbboot와 tbdn은 각각 Tibero를 기동하고 종료하는 역할을 담당한다.</p> <p>tbsvr과 tlistener 실행 파일은 반드시 tbboot 명령어를 이용하여 실행되어야 하며, 절대로 직접 실행해서는 안 된다.</p>   |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
| client/bin | <table><tr><th>유틸리티</th><th>설명</th></tr><tr><td>tbSQL</td><td>기본적인 클라이언트 프로그램으로 사용자가 직접 SQL 질의를 하고 그 결과를 확인할 수 있는 유틸리티이다.</td></tr><tr><td>T-Up</td><td>다른 데이터베이스의 내용을 Tibero의 데이터베이스로 옮기는 것을 지원하는 유틸리티이다.</td></tr><tr><td>tbExport</td><td>논리적 백업이나 데이터베이스 간에 데이터 이동을 위해 데이터베이스의 내용을 외부 파일로 저장하는 유틸리티이다.</td></tr><tr><td>tbImport</td><td>외부 파일에 저장된 내용을 데이터베이스로 가져오는 유틸리티이다.</td></tr><tr><td>tbLoader</td><td>대량의 데이터를 데이터베이스로 한꺼번에 읽어 들이는 유틸리티이다.</td></tr><tr><td>tbpc</td><td>C 언어로 작성된 프로그램 안에서 내장 SQL(Embedded SQL)을 사용하는 프로그램을 개발할 때 이를 C 프로그램으로 변환하는 유틸리티이다. 이렇게 변환된 프로그램을 C 컴파일러를 통해 컴파일할 수 있도록 도와주는 역할도 담당한다.</td></tr></table> | 유틸리티 | 설명 | tbSQL | 기본적인 클라이언트 프로그램으로 사용자가 직접 SQL 질의를 하고 그 결과를 확인할 수 있는 유틸리티이다. | T-Up | 다른 데이터베이스의 내용을 Tibero의 데이터베이스로 옮기는 것을 지원하는 유틸리티이다. | tbExport | 논리적 백업이나 데이터베이스 간에 데이터 이동을 위해 데이터베이스의 내용을 외부 파일로 저장하는 유틸리티이다. | tbImport | 외부 파일에 저장된 내용을 데이터베이스로 가져오는 유틸리티이다. | tbLoader | 대량의 데이터를 데이터베이스로 한꺼번에 읽어 들이는 유틸리티이다. | tbpc | C 언어로 작성된 프로그램 안에서 내장 SQL(Embedded SQL)을 사용하는 프로그램을 개발할 때 이를 C 프로그램으로 변환하는 유틸리티이다. 이렇게 변환된 프로그램을 C 컴파일러를 통해 컴파일할 수 있도록 도와주는 역할도 담당한다. |
| 유틸리티       | 설명  |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
| tbSQL      | 기본적인 클라이언트 프로그램으로 사용자가 직접 SQL 질의를 하고 그 결과를 확인할 수 있는 유틸리티이다.   |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
| T-Up       | 다른 데이터베이스의 내용을 Tibero의 데이터베이스로 옮기는 것을 지원하는 유틸리티이다.  |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
| tbExport   | 논리적 백업이나 데이터베이스 간에 데이터 이동을 위해 데이터베이스의 내용을 외부 파일로 저장하는 유틸리티이다.   |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
| tbImport   | 외부 파일에 저장된 내용을 데이터베이스로 가져오는 유틸리티이다.   |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
| tbLoader   | 대량의 데이터를 데이터베이스로 한꺼번에 읽어 들이는 유틸리티이다.  |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |
| tbpc       | C 언어로 작성된 프로그램 안에서 내장 SQL(Embedded SQL)을 사용하는 프로그램을 개발할 때 이를 C 프로그램으로 변환하는 유틸리티이다. 이렇게 변환된 프로그램을 C 컴파일러를 통해 컴파일할 수 있도록 도와주는 역할도 담당한다.   |      |    |       |   |      |  |          |   |          |                                     |          |                                      |      |   |

# Tibero 서버 Directory 구조

● Directory 구성

| 구분                | Data Type   |    |    |        |                           |        |                       |       |   |
|-------------------|---|----|----|--------|---------------------------|--------|-----------------------|-------|---|
| client/config     | Tibero의 클라이언트 프로그램을 실행하기 위한 설정 파일이 위치하는 디렉터리이다.   |    |    |        |                           |        |                       |       |   |
| client/include    | Tibero의 클라이언트 프로그램을 작성할 때 필요한 헤더 파일이 위치하는 디렉터리이다.   |    |    |        |                           |        |                       |       |   |
| client/lib        | Tibero의 클라이언트 프로그램을 작성할 때 필요한 라이브러리 파일이 위치하는 디렉터리이다.  |    |    |        |                           |        |                       |       |   |
| client/ssl        | 서버 보안을 위한 인증서와 개인 키를 저장하는 디렉터리이다.   |    |    |        |                           |        |                       |       |   |
| client/tbepa      | External Procedure와 관련된 설정 파일과 로그 파일이 있는 디렉터리이다.  |    |    |        |                           |        |                       |       |   |
| config            | Tibero의 환경설정 파일이 위치하는 디렉터리이다. 이 위치에 존재하는 \$TB_SID.tip 파일이 Tibero의 환경설정을 결정한다.   |    |    |        |                           |        |                       |       |   |
| database/\$TB_SID | <p>Tibero의 데이터베이스 정보를 별도로 설정하지 않는 한, 모든 데이터베이스 정보가 이 디렉터리와 그 하위 디렉터리에 저장된다. (이 디렉터리에는 데이터 자체에 대한 메타데이터 뿐만 아니라 다음과 같은 종류의 파일이 있다.)</p> <table><tr><th>파일</th><th>설명</th></tr><tr><td>컨트롤 파일</td><td>다른 모든 파일의 위치를 담고 있는 파일이다.</td></tr><tr><td>데이터 파일</td><td>실제 데이터를 저장하고 있는 파일이다.</td></tr><tr><td>로그 파일</td><td>데이터 복구를 위해 데이터에 대한 모든 변경 사항을 저장하는 파일이다.</td></tr></table> | 파일 | 설명 | 컨트롤 파일 | 다른 모든 파일의 위치를 담고 있는 파일이다. | 데이터 파일 | 실제 데이터를 저장하고 있는 파일이다. | 로그 파일 | 데이터 복구를 위해 데이터에 대한 모든 변경 사항을 저장하는 파일이다. |
| 파일                | 설명  |    |    |        |                           |        |                       |       |   |
| 컨트롤 파일            | 다른 모든 파일의 위치를 담고 있는 파일이다.   |    |    |        |                           |        |                       |       |   |
| 데이터 파일            | 실제 데이터를 저장하고 있는 파일이다.   |    |    |        |                           |        |                       |       |   |
| 로그 파일             | 데이터 복구를 위해 데이터에 대한 모든 변경 사항을 저장하는 파일이다.   |    |    |        |                           |        |                       |       |   |



# Tibero 서버 Directory 구조

● Directory 구성

| 구분  | Data Type  |    |    |  |  |  |   |   |   |                |   |
|---|--|----|----|--|--|--|---|---|---|----------------|---|
| database/\$TB_SID /psm                        | tbPSM 프로그램을 컴파일드 모드(Compiled mode)로 컴파일하는 경우, 컴파일된 파일이 저장되는 디렉터리이다. 하지만, 현재 Tibero에서는 인터프리터 모드만을 지원하고 있다. 자세한 내용은 "Tibero tbPSM 안내서"를 참고한다.  |    |    |  |  |  |   |   |   |                |   |
| instance/\$TB_SID /audit                      | 데이터베이스 사용자가 시스템 특권 또는 스키마 객체 특권을 사용하는 것을 감시(AUDIT)한 내용을 기록한 파일이 저장되는 디렉터리이다.   |    |    |  |  |  |   |   |   |                |   |
| instance/\$TB_SID /log                        | <div>Tibero의 로그 파일이 저장되는 디렉터리이다.</div> <table><tr><th>파일</th><th>설명</th></tr><tr><td>트레이스 로그 파일<br/>( Tibero6 FS06 버전<br/>부터 slog )</td><td>디버깅을 위한 파일이다. 서버가 하는 모든 일이 자세하게 기록되는 파일이며, 서버 성능이 저하되는 원인을 찾거나 Tibero 자체의 버그를 해결하는 데 사용될 수 있다.</td></tr><tr><td>DBMS 로그 파일<br/>( Tibero6 FS06 버전<br/>부터 dlog )</td><td>트레이스 로그 파일에 기록되는 정보보다 좀 더 중요한 정보가 기록되는 파일이며, 서버 기동 및 종료, DDL 문장의 수행 등이 기록되는 파일이다.</td></tr><tr><td>EVENT 로그 파일<br/>( Tibero6 FS06 버전<br/>부터 ilog )</td><td>설정된 event에 대한 트레이스 로그가 기록되는 파일이며, event 로그를 보려면 tbev를 이용해야한다.</td></tr><tr><td>LISTENER 로그 파일</td><td>Listener의 디버깅을 위한 파일이다. 리스너에서 일어난 중요한 일이 기록되는 파일이며, 리스너의 버그를 해결하는 데 사용될 수 있다.</td></tr></table> <div>트레이스 로그 파일과 DBMS 로그 파일은 데이터베이스를 사용할수록 계속 누적되어 저장된다. 또한, 전체 디렉터리의 최대 크기를 지정할 수 있으며, Tibero는 그 지정된 크기를 넘어가지 않도록 오래된 파일을 삭제한다.</div> | 파일 | 설명 | 트레이스 로그 파일<br>( Tibero6 FS06 버전<br>부터 slog ) | 디버깅을 위한 파일이다. 서버가 하는 모든 일이 자세하게 기록되는 파일이며, 서버 성능이 저하되는 원인을 찾거나 Tibero 자체의 버그를 해결하는 데 사용될 수 있다. | DBMS 로그 파일<br>( Tibero6 FS06 버전<br>부터 dlog ) | 트레이스 로그 파일에 기록되는 정보보다 좀 더 중요한 정보가 기록되는 파일이며, 서버 기동 및 종료, DDL 문장의 수행 등이 기록되는 파일이다. | EVENT 로그 파일<br>( Tibero6 FS06 버전<br>부터 ilog ) | 설정된 event에 대한 트레이스 로그가 기록되는 파일이며, event 로그를 보려면 tbev를 이용해야한다. | LISTENER 로그 파일 | Listener의 디버깅을 위한 파일이다. 리스너에서 일어난 중요한 일이 기록되는 파일이며, 리스너의 버그를 해결하는 데 사용될 수 있다. |
| 파일  | 설명   |    |    |  |  |  |   |   |   |                |   |
| 트레이스 로그 파일<br>( Tibero6 FS06 버전<br>부터 slog )  | 디버깅을 위한 파일이다. 서버가 하는 모든 일이 자세하게 기록되는 파일이며, 서버 성능이 저하되는 원인을 찾거나 Tibero 자체의 버그를 해결하는 데 사용될 수 있다.   |    |    |  |  |  |   |   |   |                |   |
| DBMS 로그 파일<br>( Tibero6 FS06 버전<br>부터 dlog )  | 트레이스 로그 파일에 기록되는 정보보다 좀 더 중요한 정보가 기록되는 파일이며, 서버 기동 및 종료, DDL 문장의 수행 등이 기록되는 파일이다.  |    |    |  |  |  |   |   |   |                |   |
| EVENT 로그 파일<br>( Tibero6 FS06 버전<br>부터 ilog ) | 설정된 event에 대한 트레이스 로그가 기록되는 파일이며, event 로그를 보려면 tbev를 이용해야한다.  |    |    |  |  |  |   |   |   |                |   |
| LISTENER 로그 파일                                | Listener의 디버깅을 위한 파일이다. 리스너에서 일어난 중요한 일이 기록되는 파일이며, 리스너의 버그를 해결하는 데 사용될 수 있다.  |    |    |  |  |  |   |   |   |                |   |

# Tibero 서버 Directory 구조

- Directory 구성

| 구분                     | Data Type  |  |
|------------------------|--|--|
| instance/\$TB_SID/log  | 파라미터   | 설명                                     |
|                        | LOG_DEFAULT_DEST   | 로그 생성 기본 경로                            |
|                        | DBMS_LOG_FILE_SIZE   | DBMS 로그 파일 하나의 최대 크기를 설정한다.            |
|                        | DBMS_LOG_TOTAL_SIZE_LIMIT  | DBMS 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다.     |
|                        | TRACE_LOG_FILE_SIZE  | 트레이스 로그 파일 하나의 최대 크기를 설정한다.            |
|                        | TRACE_LOG_TOTAL_SIZE_LIMIT   | 트레이스 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다.     |
|                        | EVENT_TRACE_FILE_SIZE  | Event 로그 파일 하나의 최대 크기를 설정한다.           |
|                        | EVENT_TRACE_TOTAL_SIZE_LIMIT   | Event 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다.    |
|                        | LSNR_LOG_FILE_SIZE   | Listener 로그 파일 하나의 최대 크기를 설정한다.        |
|                        | LSNR_LOG_TOTAL_SIZE_LIMIT  | Listener 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다. |
| instance/\$TB_SID/path | Tibero 서버에서 Spatial과 관련된 함수를 사용하기 위한 라이브러리 파일이 있는 디렉터리이다.                                      |  |
| license                | Tibero의 라이선스 파일(license.xml)이 있는 디렉터리이다. XML 형식이므로 일반 텍스트 편집기로도 라이선스의 내용을 확인할 수 있다.            |  |
| scripts                | Tibero의 데이터베이스를 생성할 때 사용하는 각종 SQL 문장이 있는 디렉터리이다. 또한, Tibero의 현재 상태를 보여주는 각종 뷰의 정의도 이 디렉터리에 있다. |  |
| scripts/pkg            | Tibero에서 사용하는 패키지의 생성문이 저장되는 디렉터리이다.   |  |

# Tibero 로그 개요

## ● 로그의 종류

- Tibero 인스턴스를 구성하는 각종 프로세스, 스레드들은 중요한 작업 이력을 로그파일에 기록한다

| 로그 파일              | 설명   |
|--------------------|--|
| 트레이스 로그 파일 (SLOG)  | 디버깅을 위한 파일이다. 서버가 하는 중요한 일이 기록되는 파일이며, 서버 성능이 저하되는 원인을 찾거나 Tibero 자체의 버그를 해결하는 데 사용될 수 있다. |
| DBMS 로그 파일 (DLOG)  | 트레이스 로그 파일에 기록되는 정보보다 좀 더 중요한 정보가 기록되는 파일이며, 서버 기동 및 종료, DDL 문장의 수행 등이 기록되는 파일이다.          |
| EVENT 로그 파일 (ILOG) | 설정된 event에 대한 트레이스 로그가 기록되는 파일이며, event 로그를 보려면 tbev를 이용해야한다.                              |
| LISTENER 로그 파일     | Listener의 디버깅을 위한 파일이다. 리스너에서 일어난 중요한 일이 기록되는 파일이며, 리스너의 버그를 해결하는 데 사용될 수 있다.              |
| CM 로그 파일           | TAC 환경에서 구동하는 CM( Tibero Cluster Manager) 동작 정보가 기록되는 파일이다.                                |

## ● 로그 경로와 크기

- \$TB\_HOME/instance/\$TB\_SID/log
- 로그별 저장 경로와 파일 크기의 제한 설정이 가능함
- 전체 디렉터리의 최대 크기를 지정할 수 있으며, Tibero는 그 지정된 크기를 넘어가지 않도록 오래된 파일을 삭제한다.

# Tibero 로그 설정

## ● 로그 경로 설정을 위한 파라미터

| 초기화 파라미터 (=> T6 FS06 일때)      | 설명  |
|-------------------------------|---|
| LOG_DEFAULT_DEST              | 로그 전체 경로( 기본값: \$TB_HOME/instance/\$TB_SID/log ) 를 설정한다.  |
| LSNR_LOG_DEST                 | 리스너 로그 경로( 기본값: \$TB_HOME/instance/\$TB_SID/log/lsnr ) 를 설정한다.  |
| TRACE_LOG_DEST => SLOG_DEST   | 트레이스 로그 경로( 기본값: \$TB_HOME/instance/\$TB_SID/log/tracelog (=> slog) ) 를 설정한다.   |
| DBMS_LOG_DEST                 | DBMS 로그 경로( 기본값: \$TB_HOME/instance/\$TB_SID/log/dbmslog (=> dlog) ) 를 설정한다.  |
| EVENT_TRACE_DEST => ILOG_DEST | 이벤트 로그 경로( 기본값: \$TB_HOME/instance/\$TB_SID/log/event (=> ilog) ) 를 설정한다.   |
| CM_LOG_DEST                   | CM 로그 경로를 설정한다.<br>- 기본값: TIBERO 6 FS04 버전까지는 \$TB_HOME/instance/\$TB_SID/log/cm<br>TIBERO 6 FS05 버전 부터 \$CM_HOME/instance/\$CM_SID/log |

## ● 로그 크기 설정을 위한 파라미터

| 초기화 파라미터  | 설명                                     |
|---|--|
| DBMS_LOG_FILE_SIZE                                    | DBMS 로그 파일 하나의 최대 크기를 설정한다.            |
| DBMS_LOG_TOTAL_SIZE_LIMIT                             | DBMS 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다.     |
| TRACE_LOG_FILE_SIZE => SLOG_FILE_SIZE                 | 트레이스 로그 파일 하나의 최대 크기를 설정한다.            |
| TRACE_LOG_TOTAL_SIZE_LIMIT => SLOG_TOTAL_SIZE_LIMIT   | 트레이스 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다.     |
| EVENT_TRACE_FILE_SIZE => ILOG_FILE_SIZE               | Event 로그 파일 하나의 최대 크기를 설정한다.           |
| EVENT_TRACE_TOTAL_SIZE_LIMIT => ILOG_TOTAL_SIZE_LIMIT | Event 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다.    |
| LSNR_LOG_FILE_SIZE                                    | Listener 로그 파일 하나의 최대 크기를 설정한다.        |
| LSNR_LOG_TOTAL_SIZE_LIMIT                             | Listener 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다. |
| CM_LOG_FILE_SIZE                                      | CM 로그 파일 하나의 최대 크기를 설정한다.              |
| CM_LOG_TOTAL_SIZE_LIMIT                               | CM 로그 파일이 저장된 디렉터리의 최대 크기를 설정한다.       |

감사합니다