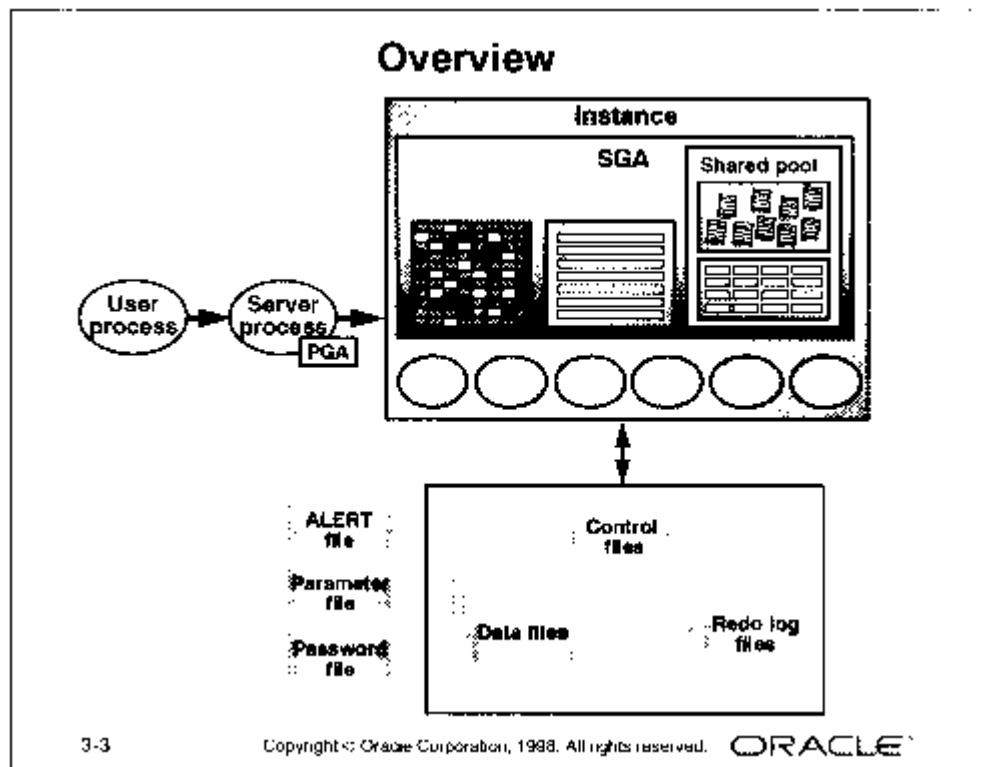


Bab 3

Mengelola Oracle Instance

Pada bab ini akan dibahas mengenai langkah-langkah administrasi database yang meliputi :

- Mengeset sistem operasi dan file password untuk autentikasi.
- Membuat file parameter.
- Men-startup instance dan membuka database.
- Menutup database dan men-shutdown instance.
- Mendapatkan dan mengeset nilai parameter.
- Mengelola session.
- Memonitor ALERT dan trace files.



Database Oracle tidak dapat dipergunakan oleh user samapai administrator database telah men-startup instance dan membuka database.

Selama database di-startup maka beberapa event berikut ini dipakai oleh database Oracle melalui tahapan :

- Start Instance
- Mount database
- Membuka Database

Setiap saat instance di startup, maka Oracle menggunakan file parameter yang berisi parameter inisialisasi untuk mengalokasikan SGA dan menjalankan background processes.

Jika instance telah dijalankan atau database telah dibuka, maka user dapat mengikuti tahapan berikut ini untuk melakukan shutdown database, yaitu :

- Menutup database
- Dismount database
- shutdown instance

Jika database ditutup maka user tidak dapat mengaksesnya. Dan untuk itu sebelum user menggunakannya maka dipastikan bahwa instance database telah aktif.

3.1. User Administrator

Ada dua tipe user database yang dapat melakukan tugas administrasi atau disebut dengan user administrator, yaitu SYS dan SYSTEM dengan privilege :

- Secara otomatis dibuatkan saat menginstall database
- Diberikan role sebagai DBA

Perbedaan dari kedua user administrator SYS dan SYSTEM ditampilkan pada tabel berikut ini :

SYS	SYSTEM
Password : change_on_install	Password : manager
Pemilik dari database data dictionary	Pemilik dari tabel internal tambahan yang dipergunakan oleh Oracle Tools

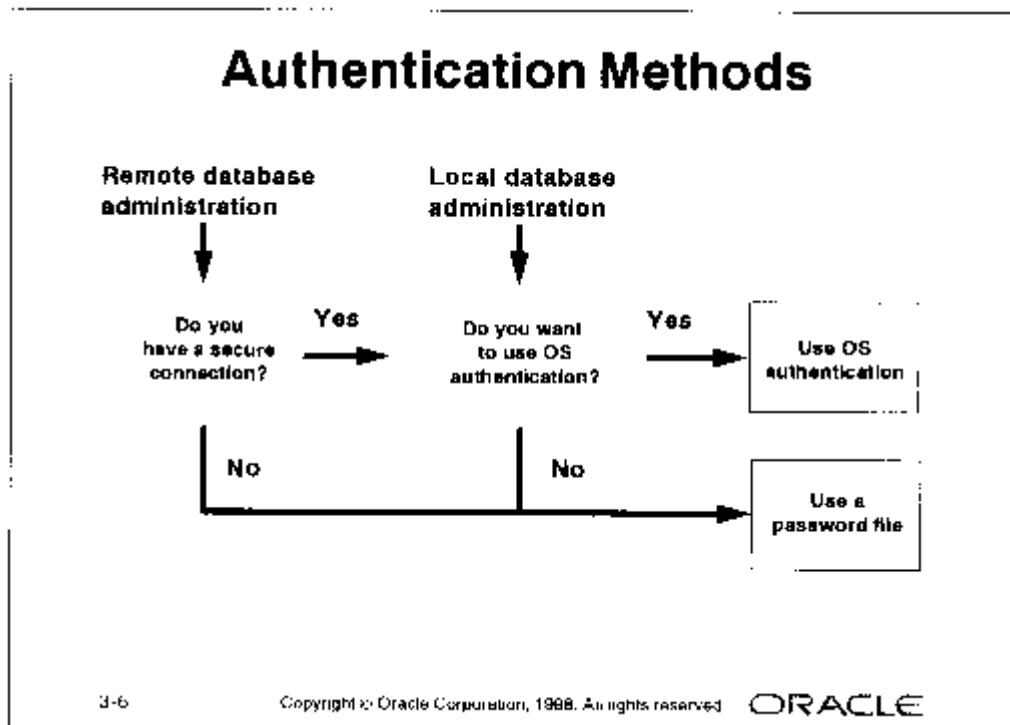
Privilege extra diperlukan untuk mengeksekusi tugas-tugas administratif dalam Oracle seperti membuat user, operasi seperti ini harus dikerjakan oleh administrator database.

Catatan : Jika perlu tidak ada salahnya untuk membuat nama user administrator tambahan untuk mengerjakan tugas administrasi sehari-hari.

3.2. Koneksi Menggunakan Privilege Administrator

Pada beberapa kasus administrator database memerlukan metode autentikasi yang khusus, karena database ada kalanya tidak dapat dibuka, khususnya untuk operasi shutdown dan startup.

Tergantung juga kepada kondisi untuk melakukan administrasi apakah database berada pada mesin PC secara lokal atau database tersimpan dalam beberapa server database dengan metode remote client, maka dapat dipilih sistem autentikasi menggunakan autentikasi sistem operasi ataupun menggunakan autentikasi administrator database.



Operating System Authentication

- **Set up the user to be authenticated by the operating system.**
- **Set REMOTE_LOGIN_PASSWORDFILE to NONE.**
- **Use the following commands to connect to a database:**

```
CONNECT / AS SYSDBA
CONNECT / AS SYSOPER
```

3-7 Copyright © Oracle Corporation, 1998. All rights reserved. ORACLE

Mengeset user dengan menggunakan autentikasi berdasarkan autentikasi sistem operasi pada sistem UNIX berbeda dari cara mengeset user menggunakan autentikasi pada sistem Windows NT/2000.

▪ UNIX

1. User harus menjadi anggota suatu group dalam UNIX, yang biasanya digunakan group *dba*, yang sudah dibuat sebelum server Oracle diinstall. Installer akan meminta privilege administrator database Oracle dan operator pada group UNIX ini.
2. Tampilkan file */etc/groups* dan */etc/passwd* untuk menentukan anggota dari group pada sistem operasi UNIX.

Pada contoh berikut ini merupakan contoh dari isi file */etc/passwd*

```
user15:x:1064:100::/home/disk3/user15:/bin/ksh
oracle:x:920:100::/export/home/oracle:/bin/ksh
nanang:x:1032:100::/users/nanang :/bin/ksh
```

Pada contoh berikut ini merupakan contoh dari isi file */etc/passwd*

```
dba::100:root,oracle,tigger,nanang
```

3. Yakinkan bahwa parameter `REMOTE_LOGIN_PASSWORDFILE` adalah kosong, yang merupakan nilai default dari parameter ini.
4. Koneksikan ke database dengan menggunakan privilege `SYSDBA` atau `SYSOPER` yang merupakan privilege khusus bagi administrator database. Perintah berikut juga berimplikasi untuk menghubungkan schema yang dimiliki oleh SYS :

```
CONNECT / AS {SYSDBA | SYSOPER}
```

▪ **Windows NT/2000**

1. Buatlah group untuk user lokal Windows NT/2000 yang baru bernama `ORA_<SID>_DBA` dan `ORA_<SID>_OPER` yang dispesifikasi pada instance, atau `ORA_DBA` dan `ORA_OPER` yang tidak dispesifikasi dalam instance.
2. Tambahkan user sistem operasi Windows NT/2000 pada group tersebut diatas. Jika dilakukan akses terhadap domain tersebut maka secara otomatis akan divalidasi dan diautorisasi sebagai DBA.
3. Yakinkan bahwa parameter `REMOTE_LOGIN_PASSWORDFILE` adalah kosong, yang merupakan nilai default dari parameter ini.
4. Koneksikan ke database dengan menggunakan privilege `SYSDBA` atau `SYSOPER` yang merupakan privilege khusus bagi administrator database. Perintah berikut juga berimplikasi untuk menghubungkan schema ygn dimiliki oleh SYS :

```
CONNECT / AS {SYSDBA | SYSOPER}
```

Catatan : Untuk koneksi ke Windows NT/2000 secara lokal, secara remote menggunakan Windows NT/2000 atau client Windows 95/98, maka Net8 harus terinstall pada sisi client dan server.

Kegunaan dari perintah `CONNECT INTERNAL` dipakai pada versi Oracle terdahulu yang telah diganti dengan sintax baru yaitu :

```
CONNECT INTERNAL/pw AS SYSDBA
```

`CONNECT INTERNAL` seterusnya akan mendukung versi Oracle yang berikutnya.

Oracle untuk seterusnya tidak lagi mendukung setting parameter DBA_AUTHORIZATION didalam registri untuk BYPASS yang memungkinkan koneksi tanpa menggunakan password.

3.3. Menggunakan Autentikasi File Password

Oracle menyediakan utilitas password yang dapat digunakan untuk menghubungkan ke server Oracle menggunakan nama dan password standart, tetapi koneksi tersebut menggunakan schema SYS daripada menggunakan nama user yang ada.

Using Password File Authentication

- **Create the password file using the password utility:**

```
$orapwd file=$ORACLE_HOME/dbs/orapwU15\  
password=admin entries=5
```
- **Set REMOTE_LOGIN_PASSWORDFILE to EXCLUSIVE or SHARED**
- **Use the following command to connect to a database:**

```
CONNECT INTERNAL/ADMIN
```

3-8 Copyright © Oracle Corporation, 1996. All rights reserved. ORACLE

Akses ke database menggunakan file password diberikan dengan perintah GRANT khusus yang diberikan pada privilege user.

Untuk menggunakan file password pada server UNIX dan Windows NT/2000 maka lakukan langkah-langkah berikut ini :

1. Buat file password menggunakan utilitas password ORAPWD pada sistem operasi UNIX dan ORAPWD80 pada sistem operasi Windows NT/2000 seperti contoh berikut :

```
orapwd80 file=<fname> password=<password> entries=<entries>
```

Dimana :

fname nama file password yang dibuat
password password untuk SYS dan INTERNAL
entries jumlah maksimum administrator database yang diperkenankan secara berlainan (distinct).

Perintah berikut ini digunakan untuk membuat file password pada sistem operasi UNIX dengan password “admin” untuk user SYS dan INTERNAL dan menerima sampai 5 buah user dengan password berbeda.

```
$orapwd file=$ORACLE_HOME/dbs/orapwUi5 password=admin entries=5
```

2. Set parameter `REMOTE_LOGIN_PASSWORDFILE` menjadi `EXCLUSIVE` atau `SHARED`.

Dimana :

EXCLUSIVE mengindikasikan bahwa hanya satu instance yang dapat menggunakan file password dan file password tersebut berisi nama selain SYS dan SYSTEM

SHARED mengindikasikan lebih dari satu instance yang dapat menggunakan file password tersebut yaitu user yang dikenali pada file password sebagai SYS dan INTERNAL.

3. Kelakukan koneksi ke database, misalnya menggunakan perintah seperti berikut ini :

```
SVRMGR>CONNECT internal/admin
```

Catatan : Pada UNIX file-file password biasanya berada pada lokasi direktory `$ORACLE_HOME/dbs` pada sistem UNIX.

Sedangkan pada Windows NT/2000 file password merupakan file yang tersembunyi (hidden) dan biasanya berada pada susunan direktory `%ORACLE_HOME%\DATABASE`.

Pada Windows NT/2000 password untuk INTERNAL adalah “oracle”, jika Oracle diinstall menggunakan opsi pada Oracle8 Enterprise Edition,

maka anda dapat mengeset password selama installasi menggunakan opsi instalasi Custom.

3.4. Mengganti Internal Password

Pergunakan utility password pada Windows NT/2000 atau UNIX untuk menghapus atau membuat file password baru.

Pergunakan utility ORADIM80 pada Windows NT/2000 dan ORAPWD pada UNIX untuk menghapus dan membuat file password baru.

```
C:\>ORADIM80 -NEW -SID sid [-INTPWD internal_pwd] [SRVC svrname]
[MAXUSER n] [STARTMODE auto, manual] [-PFILE filename]
```

Dimana :

<i>sid</i>	nama instance
<i>internal_pwd password</i>	untuk account internal
<i>svrname</i>	nama service
<i>n</i>	jumlah maximum entries pada file password
<i>auto atau manual</i>	mengindikasikan kondisi service pada saat dimulai secara manual atau otomatis
<i>filename</i>	mengijinkan file parameter non-default yang akan dipergunakan untuk mengkonfigurasi instance.

Untuk melakukan perubahan terhadap password INTERNAL, maka ikuti langkah-langkah berikut ini :

1. Hapuslah SID sesuai password yang akan dirubah :

```
C:\>ORADIM80 -DELETE -SID sid
```

Dimana sid adalah SID yang akan dihapus

2. Membuat SID lagi yang sama dan tentukan password INTERNAL yang baru :

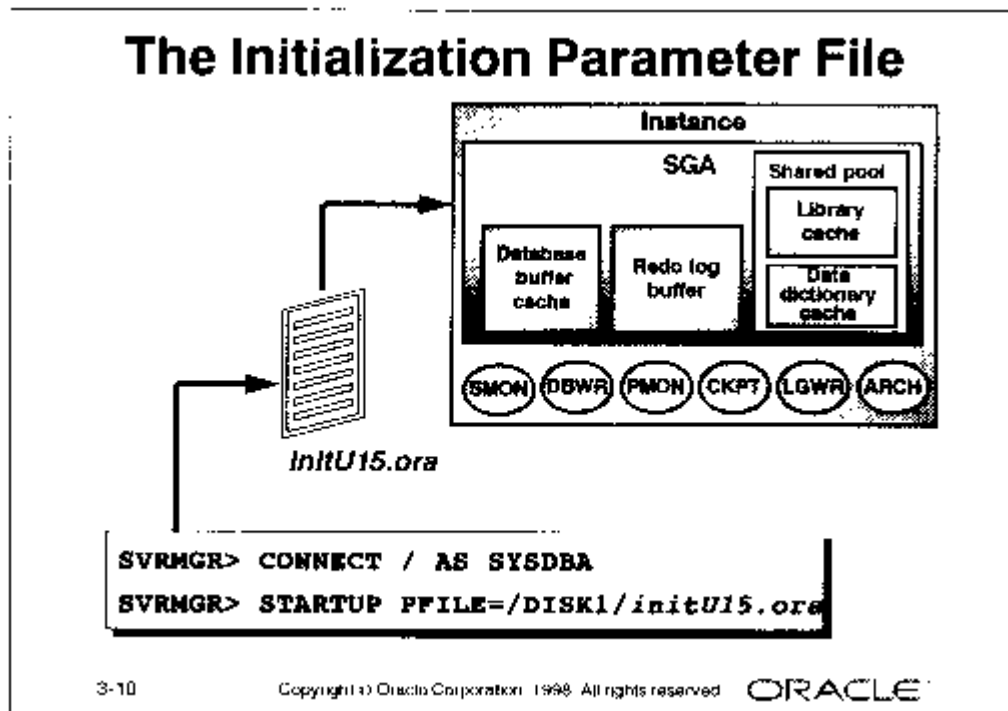
```
C:\>ORADIM80 -NEW -SID sid -INTPWD internal_pwd -MAXUSER n
```

Dimana sid adalah SID yang akan dibuat.

3.5. Membuat File Parameter

File parameter biasanya biasanya direfernsikan sebagai file `init<SID>.ora`, yang merupakan file teks yang dapat dikelola menggunakan standart editor yang disediakan sistem operasi pada umumnya.

Secar default file ini berada pada lokasi direktory `$ORACLE_HOME/dbs` pada sistem operasi UNIX dan pada direktory `%ORACLE_HOME%\database` pada Windows NT/2000.



File parameter dibaca hanya selama instance startup, dan jika file tersebut dimodifikasi maka lakukan shutdown dan restart instance untuk membuat nilai parameter yang telah dirubah menjadi berjalan efektif.

Catatan : Oracle Enterprise Manager semestinya dapat membaca file parameter untuk memulai instance database. Demikian juga untuk mengaktifkan instance atau database secara remote tetap memerlukan file parameter yang tersimpan pada mesin PC yang menjalankan Oracle Enterprise Manager. Demikian pula Oracle Enterprise Manager dapat meyimpan parameter konfigurasi yang memungkinkan anda untuk membuat beberapa

konfigurasi untuk meng-startup database tanpa perlu mencari file parameter inisialisasi. Konfigurasi yang disimpan berada didalam registri dan bukan sebagai file eksternal.

Untuk membuat dan menyimpan konfigurasi :

1. Pergunakan Instance Manager
2. Pilih menu Initialization Parameters
3. Pilih save yang terdapat pada salah satu property sheet
4. Isilah nama konfigurasi didalam Configuration property sheet.

1. Kegunaan parameter

Parameter pada file `init<SID>.ora` dapat memiliki efek yang signifikan pada performance database, dan perlu adanya modifikasi seperti beberapa cara dibawah ini untuk memproduksi sistem dengan karakteristik optimal.

- Ukuran komponen *system-global-area* (SGA) untuk mengoptimalkan performance.
- Mengeset database dan instance secara default.
- Mengeset batas database.
- Mendefinisi (pada saat pembuatan database) berbagai atribut fisik tentang database, seperti ukuran *database-block-size*.
- Menspesifikasi control files, archived log files, dan lokasi trace file.

2. Aturan Penentuan Parameter

- Menspesifikasi nilai dari format berikut : `keyword=nilai`.
- Semua parameter adalah opsional.
- Server memiliki nilai default pada setiap pengesetan parameter.
- Parameter dapat dispesifikasi untuk segala keperluan.
- Baris komentar (bagian yang tidak dikerjakan) diawali dengan simbol #.
- Menutupi setiap parameter dengan tanda petik jika digunakan karakter literal.
- File tambahan dapat disertakan menggunakan keyword `IFILE`.
- Jika ada multiple value maka dilingkupi dengan tanda parentheses (:) dan dipisahkan dengan comma.

Catatan : Selain menggunakan daftar susunan parameter yang standart dapat pula digunakan susunan secara alphabet atau per group sesuai dengan fungsinya.

Beberapa parameter yang dispesifikasi antara lain :

Parameter	Keterangan
BACKGROUND_DUMP_DEST	Lokasi dimana file trace untuk background process ditulis
COMPATIBLE	Versi dari server yang digunakan dan dengan instance yang mana seharusnya compatible. Nilai defaultnya adalah 8.0.0
CONTROL_FILES	Nama dari control files
DB_BLOCK_BUFFERS	Jumlah block yang berada pada cache didalam SGA. Nilai default dan minimum adalah 50 buffer
DB_NAME	Database mengidentifikasi sebanyak 8 karakter atau kurang. Parameter ini hanya diperlukan jika membuat database yang baru.
SHARED_POOL_SIZE	Ukuran shared pool dalam byte. Nilai defaultnya adalah 3500000.
USER_DUMP_DEST	Lokasi dimana file trace setiap user dibuat.

Beberapa parameter yang umumnya dimodifikasi antara lain :

Parameter	Keterangan
IFILE	Nama dari file parameter yang lain yang digabungkan pada file parameter saat ini. Diberikan kemungkinan untuk membuat tiga level file yang digabungkan.
LOG_BUFFER	Jumlah byte yang dialokasikan untuk buffer redo log dalam SGA.

MAX_DUMP_FILE_SIZE	Ukuran maksimum dari trace files, yang dispesifikasi sebagai jumlah block sistem operasi.
PROCESSES	Jumlah maksimum process pada sistem operasi yang dapat dikoneksi secara simultan ke instance.
SQL_TRACE	Mengeset bisa atau tidaknya fasilitas trace terhadap SQL untuk setiap sesion user.
TIMED_STATISTICS	Mengeset bisa atau tidaknya timing pada trace files dan didalam layar monitor.

3.6. Tahapan Startup dan Shutdown

Pada saat memulai database secara manual, anda dapat memilih tahapan mana yang dimulai. Pada skenario berikut ini menjelaskan perbedaan tahapan untuk memulai suatu instance.

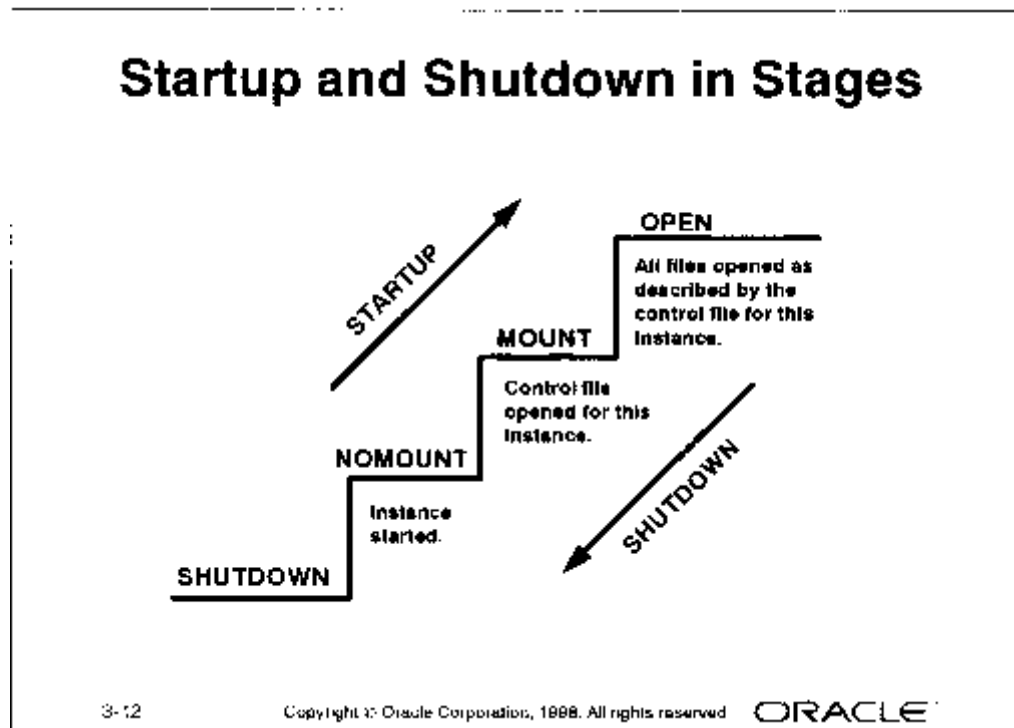
1. Memulai Instance

Biasanya jika memulai instance database tidak dilakukan mounting database, kecuali saat membuat database atau membuat kembali control file.

Memulai instance meliputi langkah-langkah berikut :

- Membaca file parameter init<SID>.ora
- Mengalokasi SGA
- Memulai background process
- Membuka file trace dan ALERT

Database mesti harus diberi nama menggunakan parameter DB_NAME pada file init<SID>.ora atau pada perintah STARTUP.



2. Mounting database

Untuk melakukan operasi perawatan yang spesifik, maka sering dilakukan langkah untuk memulai instant dan mount (menempelkan) database, tetapi tidak membuka database.

Sebagai contoh bahwa database harus di mount tetapi tidak dibuka pada saat mengalami konsidi :

- Menamai data files.
- Mengeset bisa atau tidaknya opsi pengarsipan redo log.
- Melakukan recovery database keseluruhan.

Dan sebagai contoh bahwa database harus di mount secara default pada saat mengalami konsidi :

- Menggabungkan database dengan instance yang sebelumnya dibuka.
- Mencari lokasi dan membuka control files yang dispesifikasi dalam file parameter.
- Membaca control file untuk memperoleh nama dan status dari data files dan redo log files.

3. Membuka Database

Pada operasi database secara normal berarti instance di start dan database di-mount dan dibuka, pada saat ini memungkinkan user untuk melakukan koneksi menuju database dan melakukan operasi akses data.

Pada saat membuka database terdapat beberapa kegiatan :

- Membuka data files yang online.
- Membuka redo log files yang online.

Jika terjadi keadaan bahwa redo log files tidak ada yang online pada saat starting database maka Oracle akan menampilkan pesan kesalahan.

Pada saat tahapan akhir ini, Oracle memeriksa bahwa semua redo log files yang online dapat dibuka, untuk melakukan pengecekan database secara konsisten. Jika diperlukan background process dari System Monitor (SMON) melakukan inisiatif untuk melakukan recovery terhadap instance.

4. Instance Recovery

Instance dapat mengalami kegagalan jika instance tidak dapat meneruskan untuk bekerja. Sebagai contoh jika ada crash pada sistem operasi, maka background process SMON akan secara otomatis membentuk instance recovery pada saat database dibuka kembali. Hal ini berarti bahwa redo log files yang online digunakan untuk merecovery data yang telah dicommit pada database buffer cache yang telah hilang pada saat terjadi kegagalan instance.

Recovery yang dilakukan pada instance berisi beberapa tahap berikut ini :

- Rolling forward untuk menemukan kembali data yang tidak terekam pada data files tetapi telah terekam pada redo log yang online.
- Membuka database dan menunggu sampai semua transaksi sudah di roll back sebelum membuat database yang sesuai. Semua data yang tidak terkunci dengan transaksi unrecover akan langsung dapat dipakai.
- Rolling back transaksi tak tercommit oleh SMON dan dengan server process tersendiri sebagai akses data terkunci.

5. Tahapan Shutdown

Ada tiga tahap dalam langkah shutdown instance dan database yang sedang terkoneksi dalam jaringan, yaitu :

- *Closing database*

Langkah pertama dalam shutdown database adalah menutup database. Ketika database ditutup , maka Oracle akan mencatat perubahan pada buffer cache dan isian redo log buffer cache pada redo log file yang online. Sesudah operasi ini maka Oracle akan menutup semua redo log file dan semua servis yang online. Control file masih tetap terbuka samapai database ditutup tetapi masih dalam keadaan mount.

- *Dismount database*

Langkah kedua adalah dismount database dari instance. Sesudah dismount database hanya ada instance.

Pada saat database didismount maka Oracle akan menutup control files.

- *Shutdown Instance*

Langkah terakhir pada proses shutdown database adalah shutdown instance. Ketika shutdown instance, maka file trace dan ALERT akan ditutup, dan selanjutnya SGA akan di hapus, dan background process akan berhenti.

6. Starting Up

Untuk melakukan starting up instance dipergunakan perintah berikut ini :

```
STARTUP [FORCE] [RESTRICT] [PFILE=filename] [EXCLUSIVE |  
PARALLELE | SHARED] [OPEN [RECOVER] [database] | MOUNT |  
NOMOUNT]
```

Untuk membuka database dari STARTUP NOMOUNT ke tahapan MOUNT atau dari MOUNT ke tahapan OPEN digunakan perintah ALTER DATABASE, seperti contoh :

```
ALTER DATABASE {MOUNT | OPEN}
```

Keterangan dari parameter untuk starting up diatas dapat dilihat pada tabel berikut :

Perintah	Keterangan
OPEN	Memungkinkan user untuk mengakses database
MOUNT	Melakukan mount terhadap database untuk aktifitas DBA tertentu tetapi masih belum mengijinakan user untuk mengakses database.
NOMOUNT	Membuat SGA dan menjalankan background processes tetapi masih belum mengijinakan user untuk mengakses database.
EXCLUSIVE	Mengijinkan hanya instance saat ini yang dapat mengakses database.
PARALLEL	Memungkinkan beberapa instance untuk mengakses database, dan digunakan pada Oracle Parallel Server.
SHARED	Langkah alternatif dari cara PARALLEL.
PFILE= <i>parfile</i>	Mengijinkan file parameter nondefault yang dipergunakan untuk mengkonfigurasi instance.
FORCE	Membatalkan instance yang sedang berjalan sebelum menjadi statup yang normal.
RESTRICT	Mengijinkan hanya user dengan privilege RESTRICTED SESSION yang dapat mengakses database.
RECOVER	Memulai untuk recovery media ketika database sudah berjalan.

Langkah Startup juga dapat digunakan menggunakan OEM seperti langkah berikut ini :

1. Pergunakan Instance Manager
2. Pilihlah Databse > Startup
3. Pilihlah mode startup dan berilah informasi pfilename.
4. Click OK

STARTUP Command

**Start up the instance, and
open the database.**

```
STARTUP PFILE=/DISK1/initU15.ora
```

3-13

Copyright © Oracle Corporation, 1998. All rights reserved.

ORACLE

Catatan : Database Oracle pada Windows NT/2000 berjalan seperti suatu servis yang mengabaikan penghentian proses ketika user logout. Servis harus diregistrasi oleh subsistem servis dalam Windows NT/2000.

Pada Windows NT/2000 database dapat dibuka dengan menjalankan dua servis berikut ini :

- OracleService<SID> yang dibuat oleh instance database <SID>
- Oracle Start<SID> menjalankan database secara otomatis dengan menjalankan script strt<SID>.cmd

Perlu diingat pula untuk menjalankan servis OracleTNSListener80 untuk menggunakan koneksi client-server.

Untuk menjalankan servis Oracle pada waktu startup maka pergunakan Control Panel pada dialog box servis dan pilihlah startup secara otomatis.

Pada UNIX, untuk membuat database di-startup dan di-shutdown secara otomatis dengan mengontrol sisian dari file konfigurasi sistem operasi, sebagai contoh pada directory /var/opt/oracle.

7. Troubleshooting

Pada saat mencoba untuk menggunakan Oracle Utility tanpa stating servis yang telah dijelaskan diatas akan menghasilkan pesan kesalahan seperti contoh berikut ini :

```
ORA-12547 : TNS: lost contact
```

Atau

```
ORA-09352: Windows 32-bit two-task driver unable to spawn new  
Oracle task.
```

8. Shutdown

Shutdown database untuk membuat sistem operasi offline atau untuk membackup semua struktur fisik (data) dan untuk memodifikasi parameter inisialisasi.

Untuk melakukan shutdown terhadap instance dapat dipergunakan perintah SHUTDOWN seperti ditampilkan berikut ini :

```
SHUTDOWN [ NORMAL | TRANSACTIONAL | IMMEDIATE | ABORT ]
```

Shutdown Options				
Shutdown Mode	A	I	T	N
Allow new connections	X	X	X	X
Wait until current sessions end	X	X	X	✓
Wait until current transactions end	X	X	✓	✓
Force a checkpoint and close files	X	✓	✓	✓

Shutdown mode:

A Abort	I Immediate	X NO
T Transactional	N Normal	✓ YES

3 14 Copyright © Oracle Corporation, 1998. All rights reserved. ORACLE

Langkah Shutdown juga dapat digunakan menggunakan *Oracle Enterprise Manager* (OEM) seperti langkah berikut ini :

1. Pergunakan Instance Manager
2. Pilihlah Database > Shutdown
3. Pilihlah mode shutdown
4. Click OK

- *Shutdown Normal*

Normal merupakan mode shutdown default. Database yang di-shutdown secara *normal* akan memiliki kondisi seperti berikut ini :

- Tidak ada koneksi baru yang diijinkan.
- Server Oracle akan menunggu semua user untuk disconnect sebelum shutdown secara keseluruhan.
- Oracle akan menutup dan dismount database sebelum shutdown instance.
- Pada proses startup nanti tidak memerlukan recovery instance.

- *Shutdown Transactional*

Shutdown secara transactional untuk menghindari client kehilangan data yang sedang dikerjakan. Database yang di-shutdown secara *transactional* akan memiliki kondisi seperti berikut ini :

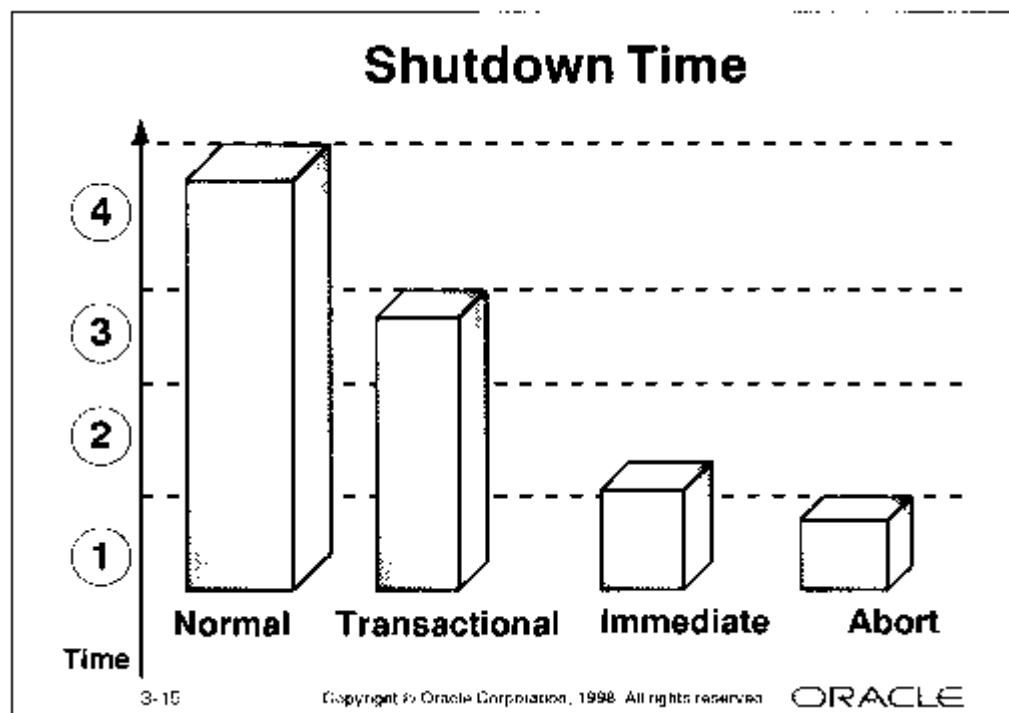
- Tidak ada client yang dapat memulai transaksi baru pada instance.
- Client akan disconnect jika client telah mengakhiri transaksi yang berlangsung.
- Jika semua transaksi telah selesai, maka akan terjadi shutdown secara langsung.
- Pada proses startup nanti tidak memerlukan recovery instance.

- *Shutdown Immediate*

Database yang di-shutdown secara *immediate* akan memiliki kondisi seperti berikut ini :

- Statement SQL yang sedang diproses oleh Oracle tidak akan diteruskan.
- Server Oracle tidak menunggu user yang sedang connect ke database untuk melakukan langkah disconnect.
- Oracle akan rollback kepada transaksi yang aktif dan akan melakukan disconnect terhadap semua user yang connect.

- Oracle akan menutup dan dismount database sebelum shutdown instance.
 - Pada proses startup nanti tidak memerlukan recovery instance.
- *Shutdown Abort*
- Jika shutdown secara *normal* dan *immediate* tidak bekerja maka dapat dilakukan pembatalan (abort) terhadap instance database yang sedang aktif. Database yang dishutdown secara *abort* akan memiliki kondisi seperti berikut ini :
- Statement SQL yang sedang diproses saat ini oleh server Oracle langsung akan dihentikan.
 - Oracle tidak akan menunggu user yang sedang connect ke database untuk melakukan disconnect
 - Transaksi yang tidak di commit tidak akan di rollback.
 - Instance langsung dihentikan tanpa menutup files.
 - Pada proses startup nanti akan memerlukan recovery instance.



Pada slide diatas ditunjukkan beberapa event ketika dilakukan perintah shutdown yang berbeda diberikan sesudah langkah ke 1 dieksekusi.

Langkah ke 1 s/d 4 menerangkan tranfer dana dari satu rekening di bank menuju yang lain.

1. Query melakukan pengecekan rekening untuk melihat neraca accoutn tersebut.
2. Mengeksekusi perintah INSERT dan DELETE untuk mentransfer dana dari rekening bank ke bank yang lain.
3. Mengeksekusi COMMIT untuk menyelesaikan transaksi secara sukses.
4. Disconnect dari Oracle.

Dengan menggunakan shutdown secara normal maka Oracle akan menunggu semua user untuk disconnect sebelum menyelesaikan shutdown.

Dengan shutdown secara transaksional maka Oracle akan menunggu sampai langkah 3 diproses, yang berarti menunggu samapai semua transaksi diselesaikan, kemusian dilakukan shutdown secara immediate.

Ketika terjadi shutdown secara immediate, maka Oracle akan menghentikan perintah SQL saat ini (dalam langkah 2) dan me-rollback transaksi yang aktif.

Ketika dilakukan shutdown abort, maka Oracle akan menghentikan perintah SQL saat ini, tetapi transaksi yang aktif tidak di-rollback.

Catatan : Pada Windows NT/2000 dapat dilakukan penutupan database dengan menghentikan service yaitu OracleService<SID> dan OracleStart<SID>.

Dengan menghentikan service dari OracleService<SID>, maka OracleStart<SID> juga akan dihentikan dan script orashut.bat akan dieksekusi.

3.7. Mendapatkan dan Mengeset Nilai Parameter

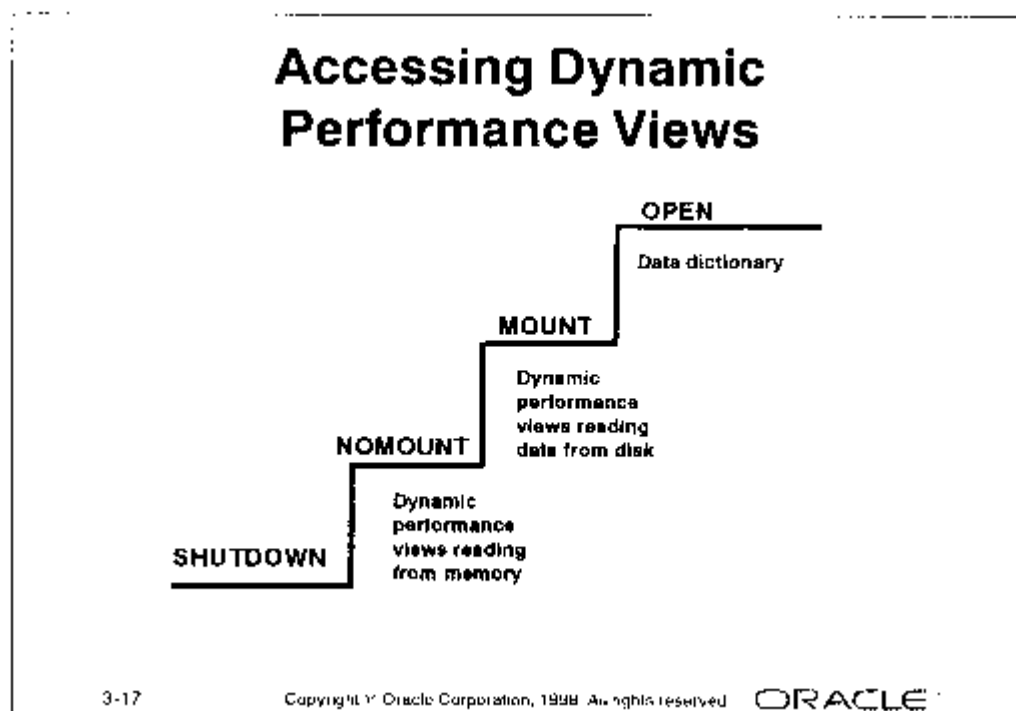
Gambaran tentang *dynamic-performance-view* adalah sebagai berikut :

- Dilakukan perawatan oleh server Oracle dan seterusnya diupdate.
- Berisi data pada disk dan memory secara terstruktur.
- Berisi data yang sangat berguna untuk keperluan performance tuning.
- Memiliki nama synonym public dengan diawali karakter V\$

View diatas perkaitan dengan penegsetan untuk mentuning performance. *Dynamic-performance-view* memiliki nama synonym diawali karakter V_\$, tetapi Oracle memiliki nama synonym diawali karakter V\$.

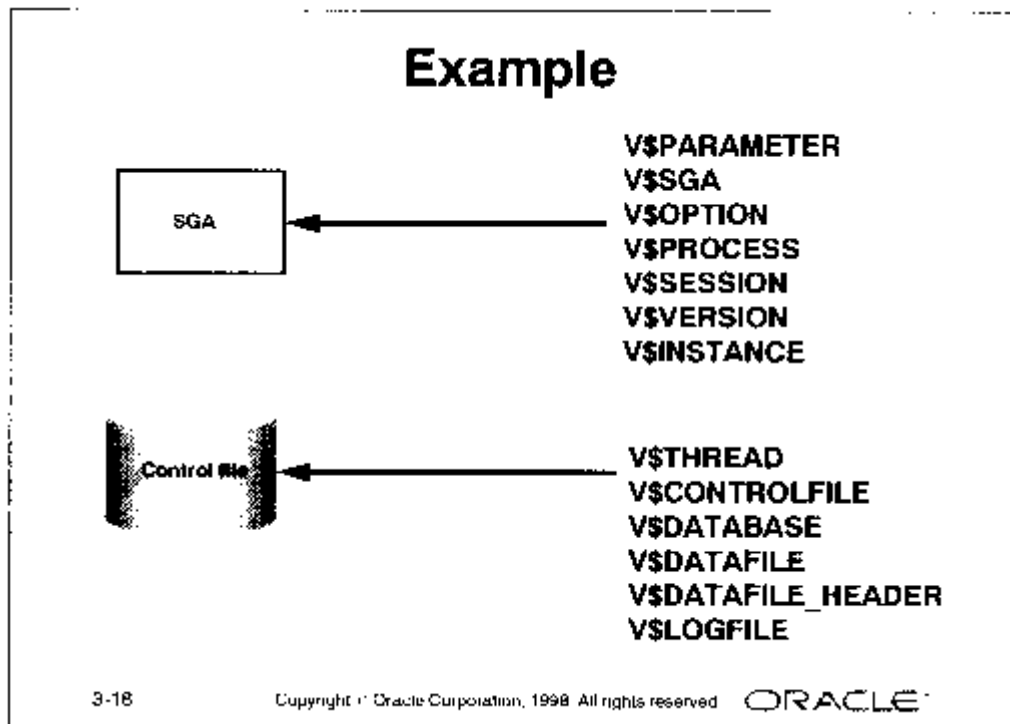
Pada saat instance diaktifkan, maka pada tahapan NOMOUNT, view V_\$ yang dapat dibaca dari memory dapat diakses. View membaca data dari control file yng diperlukan untuk database saat di-mount.

View dari V\$FIXED_TABLE akan menampilkan semua *dynamic-performance-view*.



Informasi didalam *Dynamic-performance-view* yang dapat diakses pada tahapan NOMOUNT adalah sebagai berikut :

Nama View	Keterangan
V\$PARAMETER	Berisi informasi tentang parameter inisialisasi
V\$SGA	Berisi rangkuman informasi tentang SGA
V\$OPTION	Daftar opsi yang yang diinstall pada server Oracle
V\$PROCESS	Berisi informasi tentang proses yang sedang aktif
V\$SESSION	Daftar informasi session yang sedang aktif
V\$VERSION	Daftar nomer versi dan komponennya
V\$INSTANCE	Menampilkan kondisi instance saat ini



Informasi didalam *dynamic-performance-view* yang dapat diakses pada tahapan MOUNT adalah sebagai berikut :

Nama View	Keterangan
V\$THREAD	Berisi informasi thread, contohnya tentang group redo log
V\$CONTROLFILE	Daftar nama control file. Pada kondisi NOMOUNT control file tidak berisi data.
V\$DATABASE	Berisi informasi tentang database
V\$DATAFILE	Berisi informasi data file yang berasal dari control file
V\$DATAFILE_HEADER	Menampilkan informasi header data file yang berasal dari control file
V\$LOGFILE	Berisi informasi tentang redo log file yang sedang online

3.8. Menampilkan Nilai Parameter

Untuk menentukan parameter setting bagi database yang akan diaktifkan, penggunaan perintah Server Manager yaitu SHOW PARAMETER. Perintah ini akan menampilkan semua parameter dalam urutan abjad sesuai dengan nilai parameter saat ini.

Dengan memberikan teks string sebagai opsi yang mengikuti perintah tersebut, maka akan menampilkan semua parameter yang memiliki nama opsi tersebut, sebagai contoh ditambahkan teks CONTROL maka akan menampilkan hasil sebagai berikut :

```
SVRMGR>SHOW PARAMETER control
NAME                                TYPE      VALUE
-----                                -
control_file_record_keep_time      integer 7
control_files                       string  D:\ORAWIN95\DATABASE\ctl1orcl.
```

Untuk menampilkan nilai parameter ini juga dapat digunakan *dynamic-performance-view* V\$PARAMETER untuk menentukan parameter setting yang saat ini sedang dipakai.

Displaying Current Parameter Values

- Use the Server Manager command:

```
SHOW PARAMETER control
```

- Query the dynamic performance view
V\$PARAMETER:

```
SELECT name FROM v$parameter  
WHERE name LIKE '%control%';
```

Copyright © Oracle Corporation 1998. All rights reserved. ORACLE

Jika dipergunakan menu pada *Oracle Enterprise Manager (OEM)* maka ikuti langkah berikut ini :

1. Jalankan aplikasi Instance Manager
2. Click paa Initialization parameter

3.9. Parameter Inisialisasi Dimanis

Beberapa parameter inisialisasi dibuat dinamis agar dapat dimodifikasi menggunakan perintah ALTER SESSION, ALTER SESSION, atau ALTER SESSION DEFERRED, pada saat instance tersebut sedang aktif.

Perintah yang dapat diberikan untuk memodifikasi instance dengan memberikan nilai parameter yang baru, misalnya seperti contoh berikut ini :

```
ALTER SESSION SET parameter_name = value
```

```
ALTER SYSTEM SET parameter_name =value [DEFERRED]
```

Perintah ALTER SESSION hanya memodifikasi nilai parameter untuk session yang mengeksekusi perintah tersebut.

Perintah ALTER SYSTEM merubah nilai parameter secara global. Nilai baru akan aktif samapai terjadi shutdown atau dilakukan perubahan nilai parameter lagi.

Perintah ALTER SSYSTEM DEFERRED memodifikasi nilai session seterusnya yang sedang koneksi ke database.

Statement query untuk menampilkan data dari view V\$PARAMETER atau V\$SYSTEM_PARAMETER untuk menampilkan informasi tentang parameter yang telah dimodifikasi.

```
SVRMGR> SELECT isses_modifiable, issys_modifiable,  
2> ismodified, name  
3> FROM v$system_parameter  
4> WHERE ismodified != 'FALSE';
```

```
ISSES ISSYS_MOD ISMODIFI NAME  
-----  
0 rows selected.
```

Jika pada system tersebut telah dilakukan modifikasi maka akan ditampilkan Informasi yang berisi parameter sebagai berikut :

Parameter	Jenis Informasi
ISSES_MODIFIABLE	Mengindikasikan kondisi paramenetr dapat dimodifikasi oleh ALTER SESSION.
ISSYS_MODIFIABLE	Mengindikasikan kondisi paramenetr dapat dimodifikasi oleh ALTER SYSTEM.
ISMODIFIED	Mengindikasikan bahwa modifikasi yang dilakukan oleh ALTER SESSION dengan nilai pada MODIFIED, dan modifikasi yang dilakukan oleh ALTER SYSTEM dengan nilai pada SYS_MODIFIED

Informasi pada *dynamic-performance-view* V\$PARAMETER menunjukkan nilai session saat ini, dan informasi pada *dynamic-performance-view* V\$SYSTEM_PARAMETER menunjukkan nilai system saat ini yang independet dari pada session.

Sebagai contoh jika perintah ALTER SYSTEM DEFERRED dieksekusi, maka kolom ISMODIFIED pada *dynamic-performance-view* V\$SYSTEM_PARAMETER berisi nilai MODIFIED, tetapi kolom pada *dynamic-performance-view* V\$PARAMETER akan menampilkan nilai FALSE pada session yang sama.

Catatan : Perintah ALTER SYSTEM atau ALTER SYSTEM DEFFERED memodifikasi parameter yang terekam didalam trace fiel yang disebut ALERT file.

Jika dipergunakan menu pada *Oracle Entreprise Manager* (OEM) maka ikuti langkah berikut ini :

1. Pergunakan Instance Manager
2. Pilihlah parameter yang spesifik dari daftar parameter inisialisasi
3. Masukkan nilai baru
4. Click Apply
5. Click Save untuk menyimpan perubahan yang dilakukan seperti halnya konfigurasi yang telah disimpan.

3.10. Mengelola Session

Pengsetan session yang terbatas sangat berguna, sebagai contoh jika dilakukan perawatan terhadap trukturnya atau melakukan export dan import database.

Database dapat di-startup dalam mode yang terbatas sehingga hanya sesuai bagi user yang memiliki privilege RESTRICTED SESSION.

Database juga dapat diletakkan didalam mode terbatas menggunakan perintah SQL yaitu ALTER SYSTEM.

```
ALTER SYSTEM {ENABLE | DISABLE} RESTRICTED SESSION
```

Enable and Disable Restricted Session

- Use the **STARTUP** command to restrict access to a database:

```
STARTUP RESTRICT
```
- Use the **ALTER SYSTEM** command to place an instance in restricted mode:

```
ALTER SYSTEM ENABLE RESTRICTED  
SESSION;
```

3-21 Copyright © Oracle Corporation, 1998. All rights reserved. **ORACLE**[®]

Keterangan tentang opsi yang diberikan pada perintah ALTER SYSTEM adalah sebagai berikut :

Parameter	Jenis Informasi
ENABLE RESTRICTED	Menjadikan user yang memiliki privilege RESTRICTED SESSION yang dapat login kedalam sistem database.
DISABLE RESTRICTED SESSION	Mengeset agar semua user dapat login kedalam sistem database.

Catatan : Perintah ALTER SYSTEM tidak menyebabkan disconnect ke system yang aktif, tetapi menyebabkan koneksi yang akan dilakukan nanti hanya bagi user yang memiliki privilege RESTRICTED SESSION saja.

Informasi pada *dynamic-performance-view* V\$INSTANCE berisi tentang informasi kondisi mode pembatasan tersebut, seperti contoh dibawah :

```
SVRMGR> SELECT logins FROM V$INSTANCE;  
LOGINS  
-----  
ALLOWED  
1 row selected.
```

Jika dipergunakan menu pada *Oracle Enterprise Manager (OEM)* maka ikuti langkah berikut ini :

1. Pergunakan Instance Manager
2. Pilihlah Session > Restrict
3. Atau pilih Session > Allow All.

3.11. Menghentikan Session

Jika telah menempatkan instance pada mode terbatas (restrict) maka dapat dilakukan penghentian pada semua session yang saat ini dipakai oleh user, hal ini dilakukan pada saat akan melakukan tugas-tugas pengesetan dan administrasi database.

Terminating Sessions

- 1. Identify which session to terminate with the dynamic performance view V\$SESSION:**

```
SELECT sid, serial# FROM v$session  
WHERE username='SCOTT';
```
- 2. Execute the ALTER SYSTEM command:**

```
ALTER SYSTEM KILL SESSION '7,15';
```

3-22 Copyright © Oracle Corporation, 1998. All rights reserved. ORACLE®

Untuk menghentikan session yang sedang dipergunakan user dilakukan dengan perintah :

ALTER SESSION KILL SESSION 'integer1, integer2'

Keterangan tentang opsi yang diberikan pada perintah ALTER SYSTEM untuk menghentikan session adalah sebagai berikut :

Parameter	Jenis Informasi
KILL SESSION	Menghentikan session dengan cara menentukan session mana yang di kill tergantung kepada dua nilai (integer) dari view V\$SESSION.
Integer1	Nilai dari kolom SID
Integer2	Nilai dari kolom SERIAL#

Catatan : Session ID dan nomer serial dipergunakan untuk mengidentifikasi session yang merupakan nomer yang tersendiri (tidak ada yang dobel).

Hal ini untuk meyakinkan bahwa perintah ALTER SYSTEM yang diberikan pada session saat ini sampai user logout dan session yang baru menggunakan nomer session ID yang sama, karena tidak dipakai lagi.

Jika dipergunakan menu pada *Oracle Enterprise Manager* (OEM) maka ikuti langkah berikut ini :

1. Pergunakan Instance Manager
2. Lakukan ekspansi pada Session node.
3. Pilihlah session yang diinginkan.
4. Pilih menu Session > Disconnect > Immediate.

Jika diberikan perintah ALTER SYSTEM KILL SESSION maka menyebabkan background process PMON didalam kerjanya akan melakukan beberapa langkah eksekusi sebagai brikut :

- Roolback terhadap transaksi user saat ini.

- Membebaskan penguncian terhadap tabel atau baris tertentu.
- Membebaskan (menghentikan) semua resource yang disediakan bagi user.

Untuk mengidentifikasi nilai session ID dan nomer serial dari session yang dimiliki user dapat ditampilkan dengan cara melakukan query terhadap view V\$SESSION.

- *Menghentikan Session yang aktif*

Jika session yang dipakai user sedang menjalankan perintah SQL pada server Oracle maka session tersebut dalam keadaan aktif dan jika langsung dilakukan penghentian maka transaksi akan di rollback dan seketika itu user akan menerima pesan :

```
ORA-00028: your session has been killed
```

Jika session yang dipakai user sedang melakukan aktifitas yang harus diselesaikan dan tidak dapat diinterupsi, maka server Oracle akan menunggu samapai aktifitas tersebut diselesaikan.

- *Menghentikan Session yang tidak aktif*

Jika session yang dihentikan dalam keadaan tidak aktif maka pesan ORA-00028 tidak dikirimkan kepada user, tetapi kolom STATUS pada view V\$SESSION menyatakan tanda *killed*.

Jika user mencoba untuk menggunakan lagi session yang dihentikan, maka pesan ORA-00028 akan dikirimkan dan baris pada view V\$SESSION yang berisi tanda *killed* tersebut akan dihapus.

Catatan : Ketika session dihentikan, maka server Oracle tidak menghentikan processes pada sistem operasi.

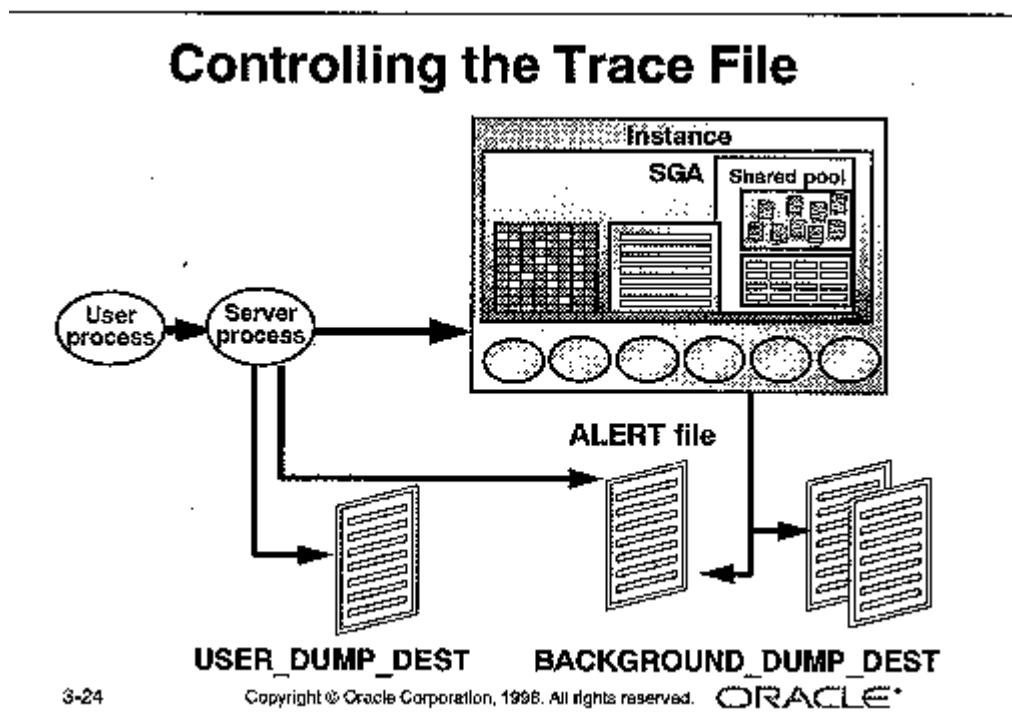
Perintah berikut ini umumnya dipergunakan pada sistem parallel server, yaitu untuk menghentikan session ketika transaksi selesai dan menghentikan server process.

```
ALTER SYSTEM DISCONNECT SESSION 'integer1, integer2'  
POST_TRANSACTION
```

3.12. File Trace dan ALERT

File trace dapat ditulis oleh server dan background processes sesuai dengan permintaan user, yang merupakan tempat untuk menyimpan semua informasi kesalahan didalam instance oracle. Jika ada error yang terdeteksi oleh background process maka informasinya akan dicatat pada trace file.

File ALERT berisi laporan tentang pesan dan kesalahan secara kronologis, jika terjadi kesalahan pada saat instance Oracle sedang berjalan maka semua pesan akan dicatat pada file ALERT. Pada saat startup database, jika file ALERT tidak ada maka Oracle akan membuatnya. User dapat mempergunakan file ALERT sebagai alternatif untuk menampilkan beberapa informasi.



Penjejukan oleh server process dapat diset enable atau disable dengan menentukan nilai TRUE atau FALSE pada parameter SQL_TRACE.

Pada statement berikut ini digunakan untuk mengaktifkan penulisan pada trace file bagi session yang sedang digunakan.

```
SQL>ALTER SESSION SET sql_trace=TRUE;
```


Keterangan tentang parameter pengontrol lokasi dan ukuran dari trace file adalah sebagai berikut :

Parameter	Jenis Informasi
BACKGROUND_DUMP_DEST	Mendefinisi lokasi dari background trace file dan ALERT file.
USER_DUMP_DEST	Mendefinisikan dimana trace file akan dibuat sesuai request user nantinya.
MAX_DUMP_FILE_SIZE	Menspesifikasi block O/S, ukuran batas dari trace file user.

Catatan : Parameter MAX_DUMP_FILE_SIZE dan USER_DUMP_DEST merupakan parameter inisialisasi yang dinamis.

Pada sistem operasi UNIX, file ALERT diberi nama *alert_<SID>.log* dan secara default diletakkan didalam direktory *\$ORACLE_HOME/rdbms/log*

Pada sistem operasi Windows NT/2000, file ALERT diberi nama *<SID>alrt.log* dan secara default diletakkan didalam direktory *%ORACLE_HOME%\RDBMS80\TRACE*.

Kesimpulan

Lakukan pengecekan terhadap file ALERT secara periodik untuk mengetahui beberapa kesalahan. Sangat penting bagi seorang administrator dalam mengecek file ALERT secara periodik untuk mendeteksi masalah yang timbul sebelum menjadi serius. Hal-hal tersebut yang meliputi :

- Mendeteksi internal error (ORA-00600) dan error akibat block data yang corrupt (ORA-01578).
- Memonitor operasi database terutama memonitor operasi yang memberikan efek kepada struktur dan parameter database dan statement Server Manager, seperti STARTUP, SHUTDOWN, ARCIVE, LOG dan RECOVER.
- Melihat parameter inisialisasi yang tidak default yang digunakan saat memulai suatu instance.

Referensi Perintah

Kontek	Referensi
Parameter Inisialisasi	DB_NAME CONTROL_FILES SHARED_POOL_SIZE BACKGROUND_DUMP_DEST DB_BLOCK_BUFFERS COMPATIBLE IFILE LOG_BUFFER PROCESSES SQL_TRACE
Parameter inisialisasi dinamis	USER_DUMP_DEST MAX_DUMP_FILE_SIZE TIMED_STATISTICS
Parameter inisialisasi dinamis (deferred)	SORT_AREA_SIZE
Dynamic Performance View	V\$FIXED_TABLE V\$PARAMETER V\$CONTROLFILE V\$DATABASE V\$DATAFILE V\$DATAFILE_HEADER V\$INSTANCE V\$LOGFILE V\$OPTION V\$PROCESS V\$PWFILERS_USERS V\$SESSION V\$SGA V\$VERSION

Kontek	Referensi
Perintah	CONNECT / AS SYSDBA CONNECT / AS SYSOPER STARTUP SHUTDOWN SHOW PARAMETER ALTER SYSTEM KILL SESSION ALTER SYSTEM DISCONNECT SESSION ... POST_TRANSACTION ALTER SYSTEM ENABLE RESTRICTED SESSION ALTER SYSTEM DISABLE RESTRICTED SESSION ALTER SESSION SET ALTER SYSTEM SET ALTER SYSTEM SET ... DEFERRED ALTER DATABASE MOUNT ALTER DATABASE OPEN