

بسمه تعالی

# Oracle DBA

ارائه دهنده :

مهندس محمد فرجی مهماندار

faraji@piau.ac.ir

www.cfscsco.ir

# سرفصل مطالب:

۱- مقدمه، معرفی ابزارها و مرور مباحث پایگاه داده ها و آشنایی با اوراکل

۲- نصب اوراکل، آشنایی با SQL\*PLUS و SQL و ارتباط CLIENT با SERVER

۳- معماری اوراکل

۴- شرح مختصری از DATA DICTIONARY

۵- تراکنش ها در اوراکل، همزمانی و مبحث LOCKING

۶- امنیت بانک اطلاعاتی (ROLE & PRIVILEGES)

۷- جامعیت بانک اطلاعاتی (CONSTRAINTS)

۸- BACKUP & RECOVERY

۹- تریگر ها

۱۰- آشنایی با PL/SQL

**جلسه اول**

**معرفی ابزارها**

**مرور مباحث پایگاه داده ها**

**آشنایی با اوراکل**

تمام کارها داده‌هایی دارند، این امر مستلزم داشتن یک روش یا مکانیزم سازمان یافته برای نگهداشت داده‌هاست. این مکانیزم تحت عنوان سیستم مدیریت بانک اطلاعاتی (DBMS) شناخته شده است. اوراکل یک سیستم مدیریت بانک اطلاعاتی می باشد که دارای سرعت، دقت، امکان دستکاری حجم بالای اطلاعات، امنیت و جامعیت اطلاعات را دارا می باشد.

### Database اوراکل را بر چه اساسی سرعت بیشتری نسبت به سایر رقبای خود دارد؟

یکی از معیارهای سنجش کارایی بانک اطلاعاتی تعداد Transaction های همزمانی است که در واحد زمان توسط آن بانک اجرا می گردد. به این قابلیت در بانک اطلاعاتی Transaction Processing performance Council (TPC) گفته می شود. اوراکل بالاترین قابلیت TPC را نسبت به سایر بانکهای اطلاعاتی دیگر دارد. بنابراین یکی از قابلیت‌های بانک اطلاعاتی اوراکل پردازش تعداد زیاد Transaction های همزمان می باشد.

\*QUERY BASE

\*OLTP(Online Transaction Proccecing)

## ریشه کلمه اوراکل چیست؟

اوراکل به معنی پیشگویی و غیب گویی و به معنای معبد یا پرستشگاه و نیز به معنای مصون از خطا و منزله از گناه می باشد.

یکی از اولین مشتریان و متقاضیان اصلی اوراکل سازمان CIA بود. Ellison و Miner پروژه ای برای سازمان CIA طراحی کردند که نام رمز ورودی آن Oracle بود. پس از آن لری الیسون و روبرت ماینر تصمیم گرفتند که محصول تجاری خود را با این نام به بازار عرضه کنند. بدین ترتیب Oracle به عنوان محصول تجاری و نام کمپانی آنها معرفی شد.

امروزه نام موتور RDBMS بانک اطلاعاتی Oracle می باشد.

## سایت‌های مربوط به اوراکل:

سایت اصلی اوراکل:

<http://www.Oracle.com/>

سایت مربوط به نظرات و سؤال و جواب‌ها:

<http://www.orafaq.com/>

سایت مربوط به منابع پیشنهادی Oracle University:

<http://www.oracle.com/education/>

سایت مربوط به داوطلب جهت آگاهی از پذیرش یا عدم پذیرش خود در امتحان OCP:

<http://www.oracle.com/education/critification/canagreement.html/>

سایت مربوط به زمان برگزاری امتحان OCP در طول سال:

<http://www.oracle.com/education/critification/testeg.html/>

## نسخه های مختلف اوراکل :

نسخه ۶: تولید در سال ۱۹۸۸، دارای قابلیت پشتیبانی از زبان برنامه نویسی PL\SQL.

نسخه ۶/۲: تولید در سال ۱۹۸۹، دارای قابلیت Parallel Server.

نسخه ۷: تولید در سال ۱۹۹۲، این نسخه از اوراکل برای سیستم عامل های Unix\Linux طراحی شد.

نسخه ۷/۱: تولید در سال ۱۹۹۴، قابل اجرا بر روی PC، ساختار اصلی اوراکل در این نسخه به روز رسانی شد.

نسخه ۸: تولید در سال ۱۹۹۷، کاربران بیشتری را تحت پوشش قرار داد، داده های سنگین تری را تبادل می کرد، قابلیت دسترسی بالاتری داشت.

نسخه ۸i: تولید در سال ۱۹۹۹، نوشته شده به زبان جاوا، (نسخه اینترنتی=i).

نسخه ۹i: تولید در سال ۲۰۰۱، دارای قابلیت Real application Server.

نسخه ۱۰g: تولید در سال ۲۰۰۴، (grid=g)، این نسخه از اوراکل تحت شبکه است.

## مرور مراحل نصب اوراکل (فایل PDF مربوطه)

### :OracleService [SID]

این سرویس مهم ترین سرویس بانک اطلاعاتی اوراکل می باشد. این سرویس وظیفه start کردن Instance بانک اطلاعاتی اوراکل را به عهده دارد. Instance مجموعه ای از ساختارهای حافظه و پردازشی است که برای دستیابی به داده های بانک اطلاعاتی مورد نیاز می باشد. زمانی می توان به یک بانک اطلاعاتی دسترسی داشت که Instance ای از آن وجود داشته باشد. بنابراین اگر این پردازش start نشود Instance بانک اطلاعاتی وجود نداشته و دستیابی به بانک نیز ممکن نخواهد بود.



به ازای هر Database ای که بر روی یک Server ایجاد می گردد یک سرویس با پیشوند Oracle Service به سرویسهای سیستم عامل اضافه می شود. می توان از تعداد این سرویسها به تعداد و نام Database های موجود بر روی کامپیوتر Server پی برد. به عنوان مثال اگر بروی یک کامپیوتر تنها یک سرویس با پیشوند Oracle Service وجود داشته باشد به معنی آن است که بر روی آن کامپیوتر تنها یک Database اوراکل وجود دارد. نامی که در انتهای این سرویس قرار دارد نام SID (System Identifier) بانک اطلاعاتی را مشخص می کند. به عنوان مثال سرویس Oracle Service ORACLE نشان می دهد که SID بانک اطلاعاتی ORACLE می باشد. این سرویسهای پارامترهای مورد نیاز برای ایجاد Instance را از فایل پارامتری Init.ora و یا Spfile[SID] به دست می آورد.

## :TNSListener

این پردازش وظیفه باز کردن شماره درگاه 1521 را بر روی ماشین Server به عهده دارد. این پردازش پارامترهای مورد نیاز خود را از فایل پارامتری Listener.ora به دست می آورد. فایل Listener.ora همواره در مسیر ORACLE-HOME\Network \admin قرار دارد. اگر این پردازش Start نشود و یا فایل پیکربندی Listener.ora وجود نداشته باشد دستیابی به بانک اطلاعاتی اوراکل ممکن نمی باشد. زیرا شماره درگاه 1521 که یکی از پارامترهای مورد نیاز برای پیکربندی SQL\*Net می باشد وجود ندارد. بنابراین وجود این پردازش برای برقراری ارتباط با بانک ضروری است.

همچنین از فایل اجرایی LSNRCTL.EXE که در مسیر ORACLE-HOME/bin قرار دارد می توان start و یا stop کردن پردازش Listener استفاده کرد. این فایل اجرایی از خط فرمان سیستم عامل اجرا می گردد. در سیستم عامل ویندوز پردازش Listener به صورت خودکار اجرا می گردد اما در سیستم عامل لینوکس و یا یونیکس این سرویس باید به صورت دستی اجرا

## فایل های پیکربندی اوراکل :

TNSNAME.ORA-۱

LISTENER.ORA-۲

SPFILE.ORA -۳

PWD.ORA -۴

## **:TNSnames.ora**

این فایل پیکربندی اطلاعات مربوط به تمامی Net Service Name های ایجاد شده که برای برقراری ارتباط با بانک اطلاعاتی در پیکربندی SQL\*Net مورد نیاز می باشد را نگهداری می کند. برای تعریف Net Service Name از ابزارهای زیر استفاده می شود:

۱- برنامه گرافیکی Net Manager

۲- برنامه گرافیکی Nat Configuration Assistant

۳- Notepad

هر Net Service Name از ۴ مشخصه زیر تشکیل می شود:

۱- نام کامپیوتر Server و یا IP آن

۲- پروتکل ارتباطی که عمدتاً TCP/IP در نظر گرفته می شود.

۳- شماره درگاه Listener که عمدتاً 1521 در نظر گرفته می شود.

۴- نام بانک اطلاعاتی که می توان از SID و یا Global Database Name بانک استفاده کرد.

## تعریف Net Service Name:

در سیستم عامل ویندوز بعد از نصب اوراکل از قسمت

Start -> program-> Oracle-OraHome 10g-> Configuration and Migration Tools-> NET manager

برنامه NET manager را اجرا می کنیم . این برنامه که برای پیکربندی معماری

Client/Server اوراکل مورد استفاده قرار می گیرد هم بر روی کامپیوتر Server و هم بر روی

کامپیوتر Client بعد از نصب اوراکل نصب می گردد.

محیط گرافیکی NET manager وظیفه ذخیره Net Service Name های ایجاد شده در فایل

TNSnames.ora را به عهده دارد. هر Service name شامل مشخصه ای است که بانک

اطلاعاتی مقصد را مشخص می کند. فایل TNSname.ora که مخفف Transparency

Network Service names می باشد همواره در مسیر ORACLE-

HOME\network\admin قرار دارد.

## پیکربندی Service Naming:

جهت برقراری ارتباط با یک بانک اطلاعاتی اوراکل به ۴ پارامتر زیر نیاز داریم:

○ نام کامپیوتر Server یا IP آن

○ پروتکل ارتباطی

○ شماره Port

○ نام بانک اطلاعاتی

اگر هر کدام از پارامترهای فوق به درستی تعریف نشود امکان برقراری ارتباط با بانک اطلاعاتی وجود ندارد از محیط NET manager و یا NET Configuration Assistant امکان تعریف Net service name های جدید و ذخیره اطلاعات مربوط به آنها در فایل TNSnames.ora وجود دارد.

## **:Listener.ora**

Listener پردازشی است بر روی Server جهت گوش دادن به درخواست‌های ارتباطی از طرف Client به Server و مدیریت ترافیک Server. هر گاه یک Client (یا یک Server همانند یک Client) درخواست ارتباط با Server را داشته باشد یک Listener ارتباط را به عهده دارد. اگر اطلاعات Client منطبق بر اطلاعات Listener باشد آنگاه Listener مجوز ارتباط با Server را صادر می‌نماید. به ازای هر پروتکل ارتباطی یک Listener جهت برقراری ارتباط با Server را صادر می‌نماید.

## **:PWD[SID].ora**

این فایل که password نامیده می شود نام و کلمه عبور تمام کاربرانی که دارای مجوز SYSDBA و یا SYSOPER می باشند را نگهداری می کند. این فایل همواره در مسیر ORACLE\_HOME/database قرار دارد. SYSDBA و SYSOPER مجوزی در بانک اطلاعاتی است که مدیریت بر روی Instance بانک اطلاعاتی جهت start ، shutdown و تغییر پارامترهای آن را به کاربران واگذار می کند. در Oracle 10g تنها کاربر SYS به صورت پیش فرض دارای مجوز SYSDBA می باشد. مهم ترین ویژگی مجوز SYSDBA و SYSOPER آن است که با داشتن این مجوز می توان حتی به یک بانک اطلاعاتی shutdown وصل شده و بانک اطلاعاتی را در وضعیت مطلوب قرار داد. با اعطای مجوز SYSDBA و SYSOPER به سایر کاربران بانک اطلاعاتی نام و کلمه عبور آن کاربر در این فایل password قرار می گیرد. دستور واگذاری مجوز SYSDBA و SYSOPER به صورت زیر است:

```
GRANT SYSDBA TO username;
```

```
GRANT SYSOPER TO username;
```



## **:Spfile[SID].ora**

هر Instance باید یک فایل پارامتری که آن را INIT.ORA می نامند برای تنظیم پارامترها و وضعیت بانکی اطلاعاتی خود داشته باشد. این فایل پارامتری در زمان Start شدن Instance توسط ابزارهای مدیریتی خوانده می شود بعد از آن که Instance ، start شد این فایل پارامتری دیگر مورد نیاز نمی باشد تا دفعه بعد که Instance دوباره start می شود . این فایل پارامتری باید بر روی کامپیوتری قرار داشته باشد که ابزارهای مدیریتی Instance در آنجا قرار گرفته اند. فایل پارامتری INIT.ORA برای مشخص کردن موارد زیر مورد استفاده قرار می گیرد:

- میزان فضای قابل تخصیص به ساختار حافظه ای اوراکل
- Rollback Segment هایی که در Instance مورد استفاده قرار می گیرد.
- تنظیمات سایر زبانهایی که پشتیبانی می شود.
- تنظیمات اوراکل برای حالت Parallel Server .
- بانک اطلاعاتی و Control File های آن که مورد استفاده هستند.

## PFILE Example

```
# Initialization Parameter File: initdb01.ora
db_name                = db01
instance_name          = db01
control_files           = ( /u03/oradata/db01/control01db01.ctl,
                           /u03/oradata/db01/control02db01.ctl)

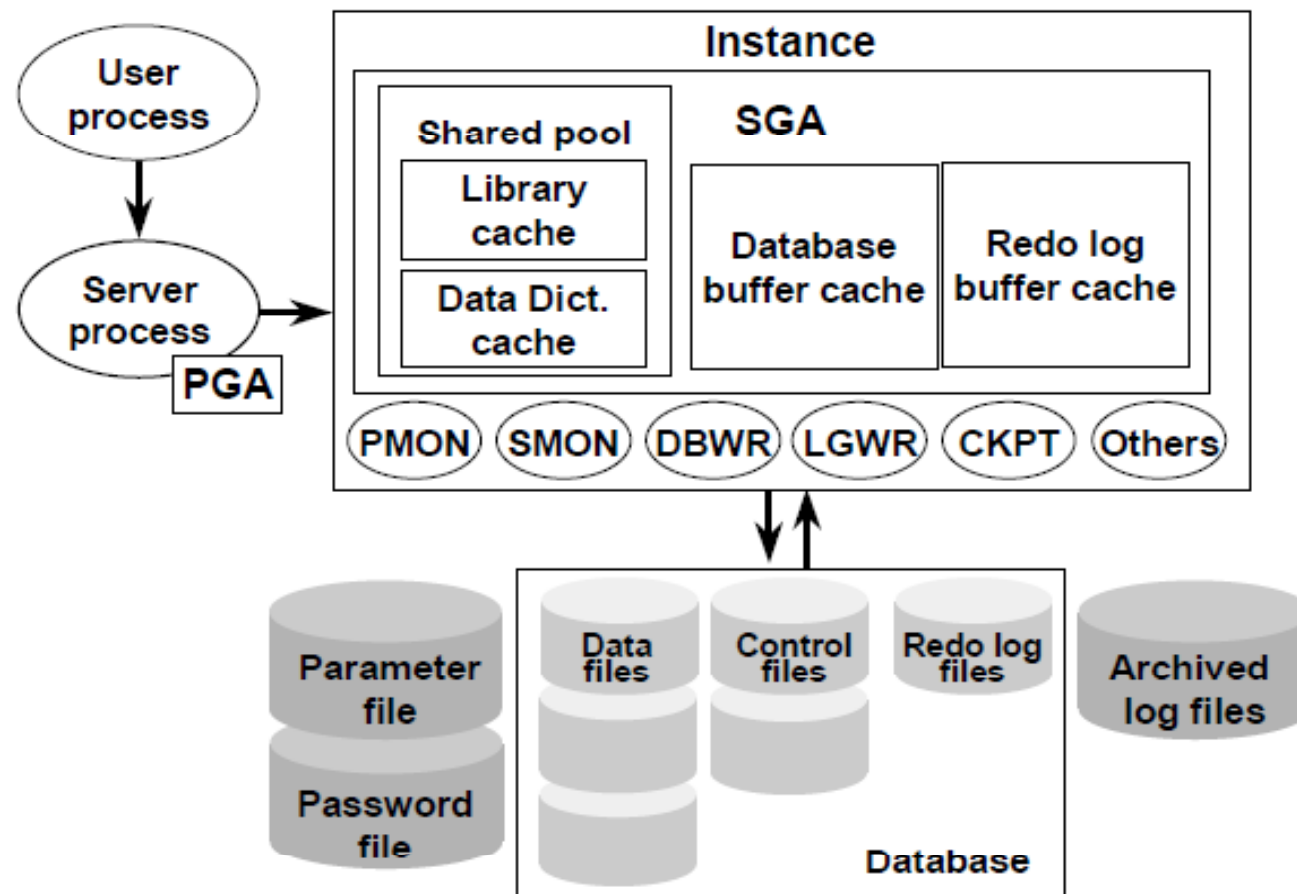
db_block_size           = 4096
db_block_buffers        = 500
shared_pool_size        = 31457280  # 30M Shared Pool
db_files                = 1024
max_dump_file_size      = 10240
background_dump_dest    = /u05/oracle91/admin/db01/bdump
user_dump_dest          = /u05/oracle91/admin/db01/udump
core_dump_dest          = /u05/oracle91/admin/db01/cdump
undo_management          = auto
undo_tablespace         = undtbs
. . .
```

**جلسه دوم**

**معماری بانک اطلاعاتی اوراگل**

## مولفه های بانک اطلاعاتی اوراکل

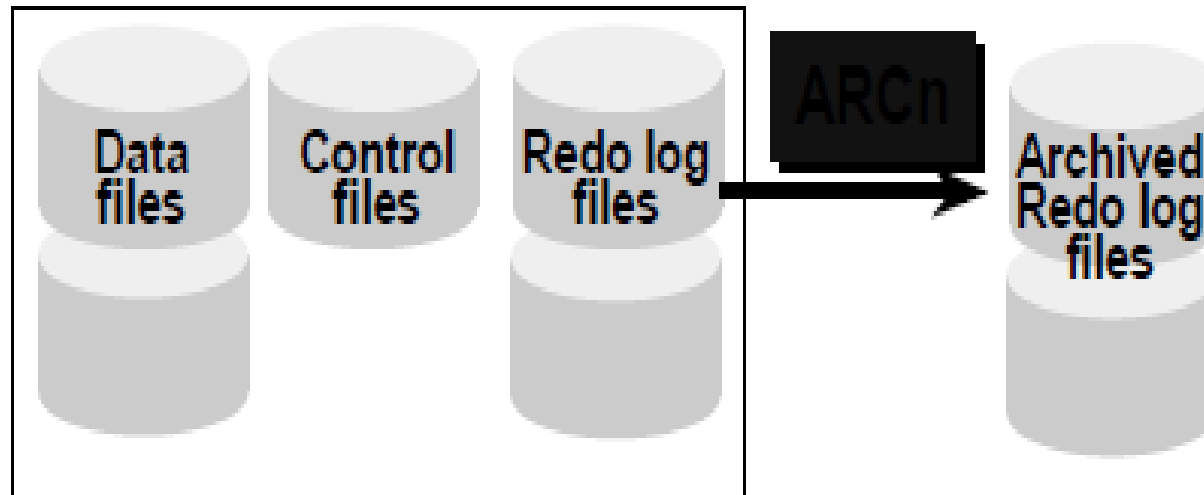
### Overview of Primary Components



## معماری اوراکل :

بانک اطلاعاتی اوراکل شامل یک لایه منطقی و یک لایه فیزیکی است. لایه فیزیکی شامل فایل هایی است که در دیسک قرار دارند و لایه منطقی لایه ای است که عناصر آن داده ها را به لایه فیزیکی منتقل می کنند

**لایه فیزیکی اوراکل از سه فایل زیر تشکیل شده است :**

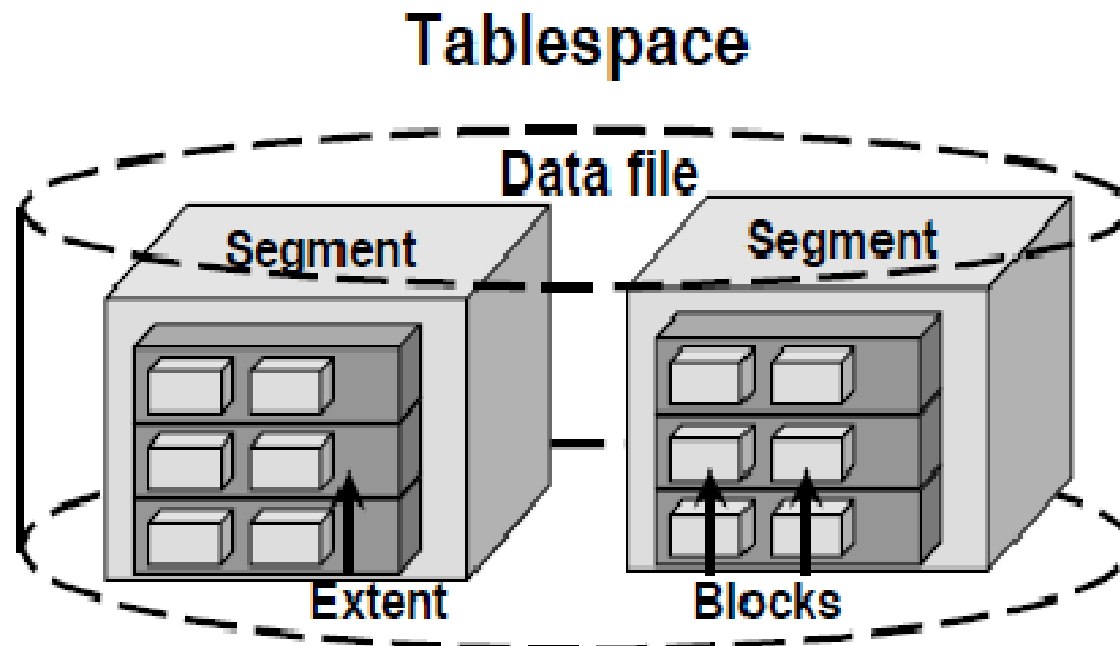


## لایه منطقی اوراکل از عناصر زیر تشکیل شده است :

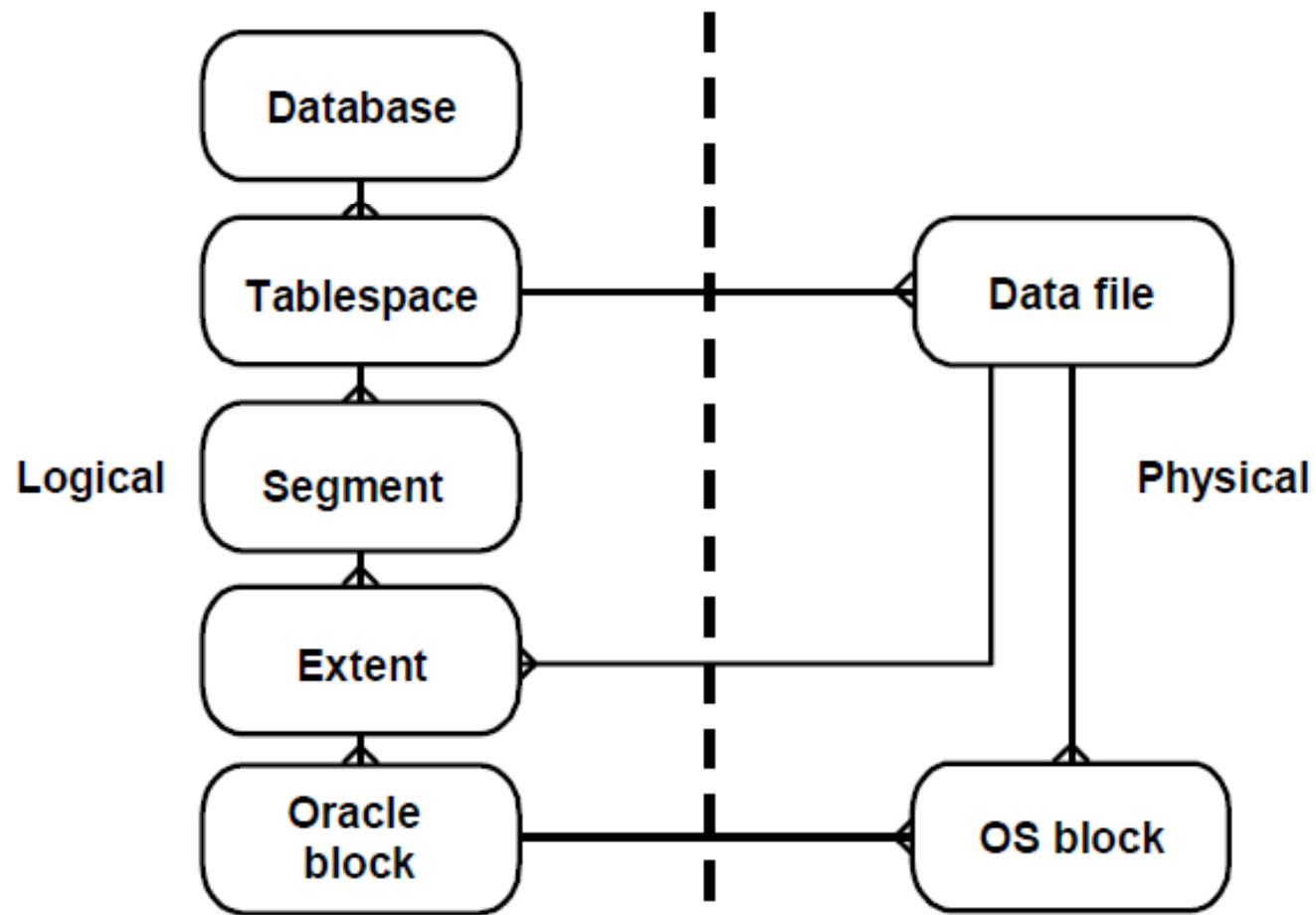
۱- یک یا چند Tablespace

۲- اشیاء بانک اطلاعاتی شامل جدول ، ویو ، شاخص و....

**وظیفه Tablespace ها در بانک اطلاعاتی اوراکل به هم پیوسته کردن فضای فیزیکی ناپيوسته Data File ها و ایجاد یک فضای فیزیکی پیوسته بزرگتر برای بانک اطلاعاتی است .**



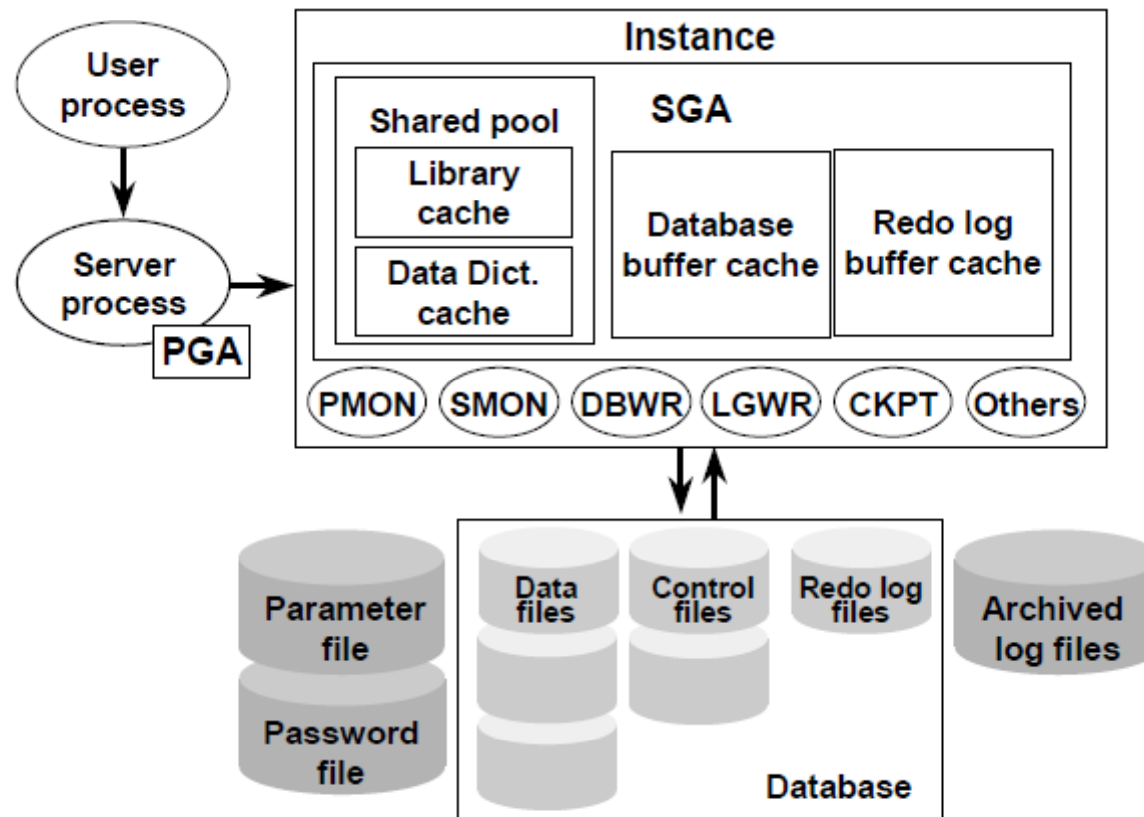
## Database Storage Hierarchy



# : INSTANCE

به مجموعه ای از پردازش ها و ساختارهای حافظه که برای دسترسی به داده های بانک اطلاعاتی نیاز می باشد Instance گفته می شود .

## Overview of Primary Components





## :OracleService [SID]

این سرویس مهم ترین سرویس بانک اطلاعاتی اوراق می باشد. این سرویس وظیفه start کردن Instance بانک اطلاعاتی اوراق را به عهده دارد. Instance مجموعه ای از ساختارهای حافظه و پردازشی است که برای دستیابی به داده های بانک اطلاعاتی مورد نیاز می باشد. زمانی می توان به یک بانک اطلاعاتی دسترسی داشت که Instance ای از آن وجود داشته باشد. بنابراین اگر این پردازش start نشود Instance بانک اطلاعاتی وجود نداشته و دستیابی به بانک نیز ممکن نخواهد بود.

به ازای هر Database ای که بر روی یک Server ایجاد می گردد یک سرویس با پیشوند Oracle Service به سرویسهای سیستم عامل اضافه می شود. می توان از تعداد این سرویسها به تعداد و نام Database های موجود بر روی کامپیوتر Server پی برد. به عنوان مثال اگر بروی یک کامپیوتر تنها یک سرویس با پیشوند Oracle Service وجود داشته باشد به معنی آن است که بر روی آن کامپیوتر تنها یک Database اوراکل وجود دارد. نامی که در انتهای این سرویس قرار دارد نام SID (System Identifier) بانک اطلاعاتی را مشخص می کند. به عنوان مثال سرویس Oracle Service ORACLE نشان می دهد که SID بانک اطلاعاتی ORACLE می باشد. این سرویسهای پارامترهای مورد نیاز برای ایجاد Instance را از فایل پارامتری Init.ora و یا Spfile[SID] به دست می آورد.

## System Global Area (SGA)

- SGA is dynamic and sized using `SGA_MAX_SIZE`.
- SGA memory allocated and tracked in granules by SGA components
  - Contiguous virtual memory allocation
  - Size based on `SGA_MAX_SIZE`

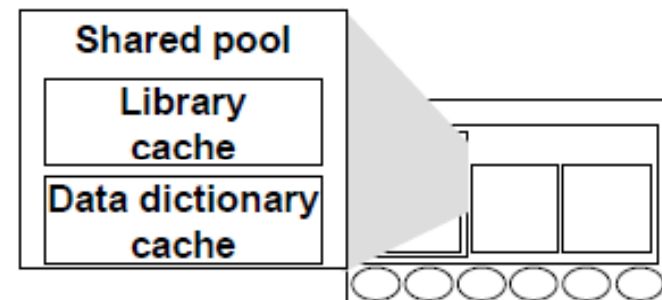
ناحیه سراسری سیستم بوده که بر اساس پارامتر `SGA_MAX_SIZE` میزان اندازه آن مشخص می شود

## Shared Pool

The shared pool is used to store the most recently executed SQL statements and the most recently used data definitions.

- It consists of two key performance-related memory structures:
  - Library cache
  - Data dictionary cache
- Sized by the parameter `SHARED_POOL_SIZE`.

```
ALTER SYSTEM SET  
SHARED_POOL_SIZE = 64M;
```



## ویژگی های Library Cash :

۱- برای SHARE کردن دستورات مشترک SQL و PL/SQL مورد استفاده قرار می گیرد .

۲- توسط الگوریتم LRU مدیریت می شود.

۳- بدون نیاز به RESTART شدن INSTANSE اندازه آن قابل تغییر است.

## ویژگی های Dictionary Cash :

۱- این ناحیه دارای اطلاعاتی شامل : نام کاربران، مجوزهای کاربران ، نام ومحل فیزیکی جداول و... می باشد که در زمان PROCEDURAL شدن یک درخواست NON-PROCEDURAL مورد استفاده قرار می گیرد.

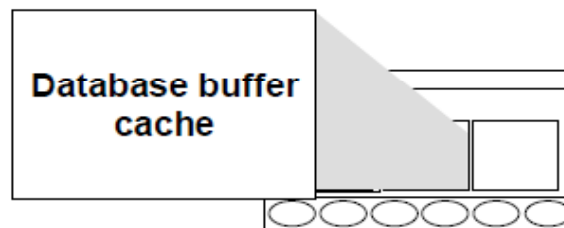
۲- افزایش کارایی بانک اطلاعاتی را با کاهش مراجعه به DATA DICTIORARY بر عهده دارد

۳- بدون نیاز به RESTART شدن INSTANSE اندازه آن قابل تغییر است.

## Database Buffer Cache

The database buffer cache stores copies of data blocks that have been retrieved from the data files.

- It enables great performance gains when you obtain and update data.
- It is managed through a least recently used (LRU) algorithm.
- `DB_BLOCK_SIZE` determines the primary block size.



ORACLE

### ویژگی ها :

- ۱- برای افزایش کارایی بانک اطلاعاتی در انجام `UPDATE` , `QUERY` داده مورد استفاده قرار می گیرد.
- ۲- توسط الگوریتم `LRU` مدیریت می شود.
- ۳- بدون نیاز به `RESTART` شدن `INSTANCE` با `ALTER SYSTEM` اندازه آن قابل تغییر است.

## Database Buffer Cache

- Consists of independent sub-caches:
  - DB\_CACHE\_SIZE
  - DB\_KEEP\_CACHE\_SIZE
  - DB\_RECYCLE\_CACHE\_SIZE
- Database buffer cache can be dynamically resized to grow or shrink using ALTER SYSTEM.

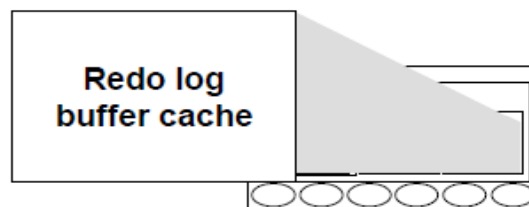
```
ALTER SYSTEM SET DB_CACHE_SIZE = 96M;
```

- DB\_CACHE\_ADVICE can be set to gather statistics for predicting different cache size behavior.

## Redo Log Buffer Cache

The redo log buffer cache records all changes made to the database data blocks.

- Its primary purpose is recovery.
- Changes recorded within are called redo entries.
- Redo entries contain information to reconstruct or redo changes.
- Size is defined by LOG\_BUFFER.



ORACLE

ویژگی ها :

۱ - اولین مولفه RECOVERY بانک اطلاعاتی است .

۲ - تمامی تغییراتی که از طریق دستورات INSERT , UPDATE, DELETE , ALTER , DROP

CREATE در بانک اطلاعاتی اعمال می شود از طریق این ناحیه از حافظه به فایل های فیزیکی REDO LOG

FILE فرستاده می شود.



## Large Pool

The large pool is an optional area of memory in the SGA configured only in a shared server environment.

- It relieves the burden placed on the shared pool.
- This configured memory area is used for session memory (UGA), I/O slaves, and backup and restore operations.
- Unlike the shared pool, the large pool does not use an LRU list.
- Sized by `LARGE_POOL_SIZE`.

```
ALTER SYSTEM SET LARGE_POOL_SIZE = 64M;
```

ORACLE

ویژگی ها :

۱- نگهداری اطلاعات SESSION کاربران

۲- انجام BACKUP

۳- RESTORE کردن داده ها

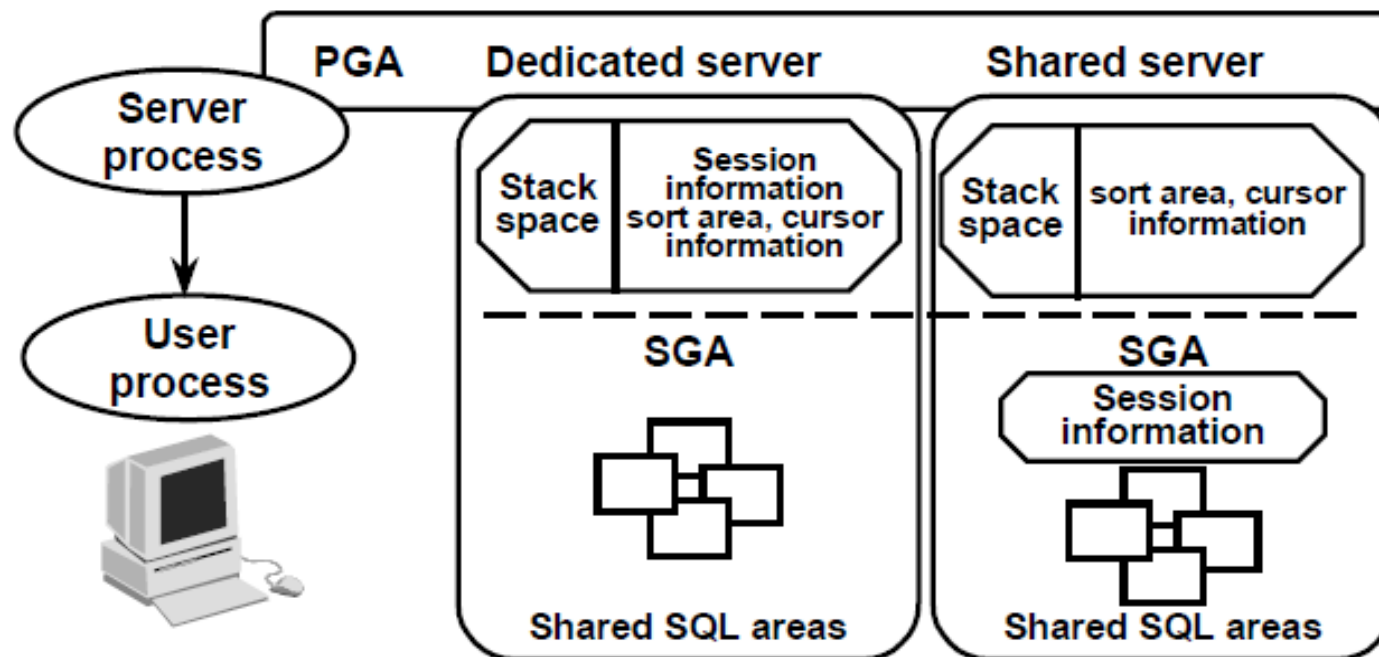
## Java Pool

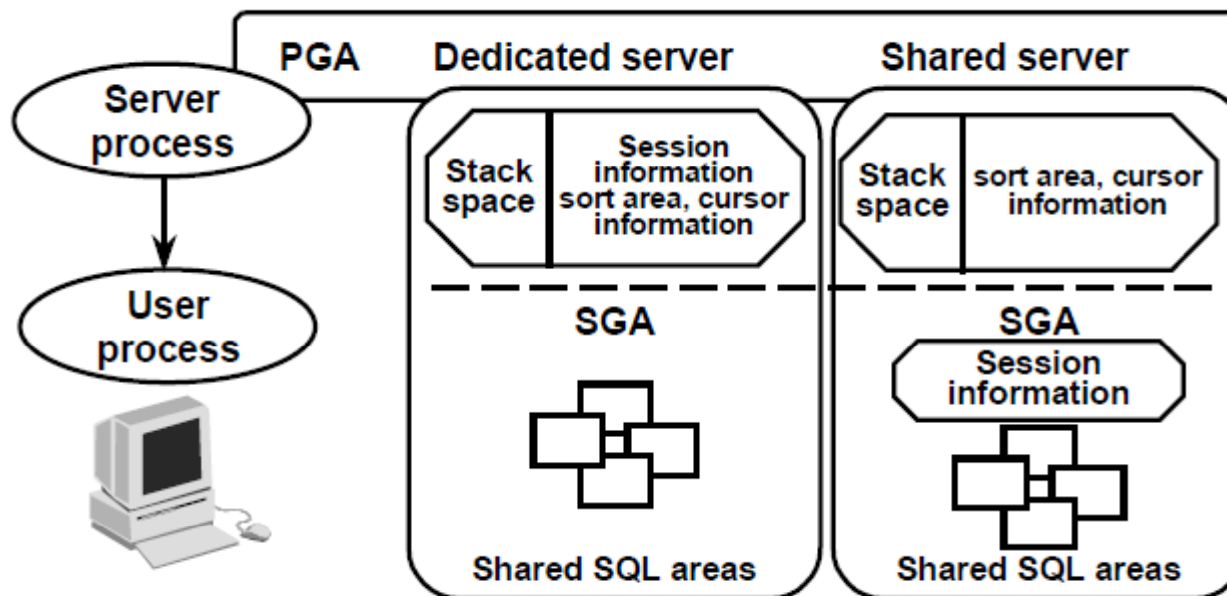
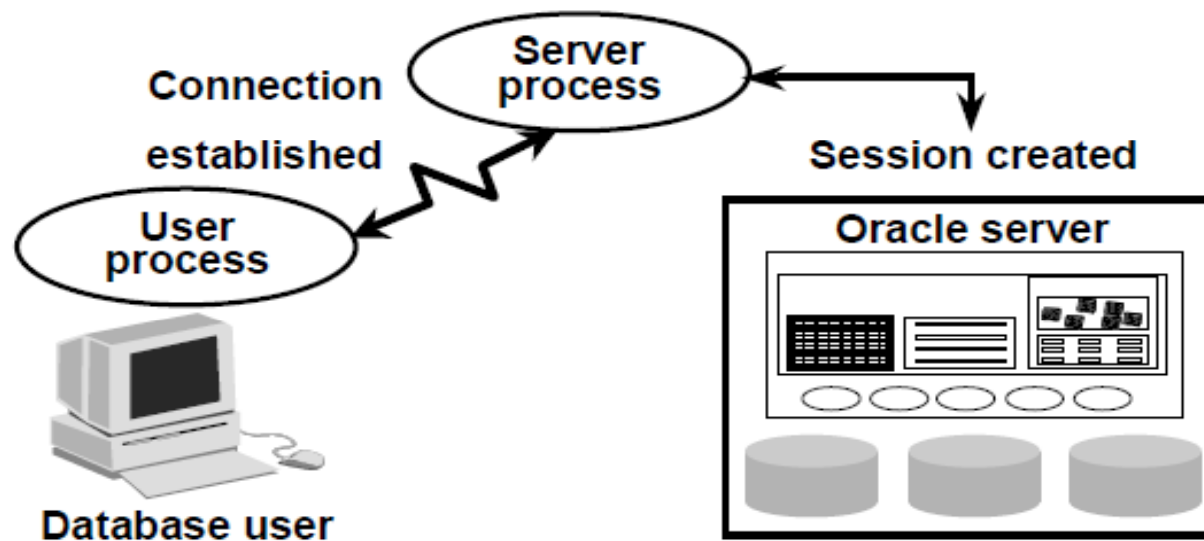
The Java pool services the parsing requirements for Java commands.

- Required if installing and using Java.
- It is stored much the same way as PL/SQL in database tables.
- It is sized by the `JAVA_POOL_SIZE` parameter.

## Program Global Area (PGA)

The PGA is memory reserved for each user process that connects to an Oracle database.





## مراحل اجرای یک QUERY مانند `SELECT * FROM EMP` :

۱- ارسال درخواست از USER PROCESS به SERVER PROCESS بانک اطلاعاتی

۲- دریافت درخواست کاربر توسط SERVER PROCESS

۳- جستجو در ناحیه SGA توسط SERVER PROCESS برای وجود TRACE قبلی از اطلاعات ، همچنین وجود یا عدم وجود جداول EMP

۴- اگر اطلاعات درخواست شده از قبل در ناحیه SGA وجود داشته باشد، بهترین حالت بوده و نیاز به انجام عملیات I/O در DATA FILE , DATA DICTIONARY های بانک اطلاعاتی وجود ندارد.

۵- اگر اطلاعات درخواست شده در ناحیه SGA وجود نداشته باشد انگاه مراحل زیر اتفاق می افتد :

- خواندن اطلاعات مورد نیاز از DD و قرار دادن آن در ناحیه DD CASHE از SHARED POOL
- ایجاد یک TRACE از دستور که شامل PARSE TREE و EXECUTION PLAN اجرایی آن دستور می باشد
- قرار دادن نسخه ای از TRACE دستور در ناحیه LIBRARY CASHE از SHARED POOL
- خواندن بلوک های داده ای از DATA FILE و قرار دادن آن در ناحیه BATABASE BUFFER CASH

۶- ارسال محتوی بلوک های داده ای خوانده شده به کاربر درخواست کننده.

## مراحل اجرای دستور INSERT مانند INSERT INTO EMP :

۱- ارسال درخواست از USER PROCESS به SERVER PROCESS بانک اطلاعاتی

۲- دریافت درخواست کاربر توسط SERVER PROCESS

۳- قرار گرفتن دستور INSERT در ناحیه REDO LOG BUFFER در SGA

۴- جستجو در ناحیه SGA توسط SERVER PROCESS برای وجود TRACE قبلی از اطلاعات ، همچنین وجود یا عدم وجود جداول EMP

۵- اگر اطلاعات درخواست شده از قبل در ناحیه SGA وجود داشته باشد، این اطلاعات شامل نام جدول ، نام ستونها ، مجوز ها، ایندکس ها و constraint ها می باشد در آن صورت مراحل زیر اتفاق می افتد :

- بلوک داده ای کاندید برای INSERT در ناحیه SGA در DATA BLOCK BUFFER CACHE قرار می گیرد

- محتوی درون بلوک داده ای تغییر می کند به طوری که این رکورد جدید در آن درج می گردد

- تصویر قبلی بلوک داده ای در ROLLBACK SEGMENT قرار می گیرد.

- HEADER های DATA FILE و DATA BLOCK طوری تغییر می کند که تغییرات انجام شده را نشان بدهد.

۵- اگر اطلاعات درخواست شده در ناحیه SGA وجود نداشته باشد در آن صورت ابتدا مراحل PARSE TREE و EXECUTION PLAN اتفاق می افتد و کپی از آن در ناحیه SHARED POOL قرار می گیرد و سایر مراحل مانند مرحله ۵ انجام می گیرد.

۶- با توجه به اینکه دستور INSERT یک TRANSACTION در بانک اطلاعاتی است طی دو حالت زیر خاتمه می یابد :

• COMMIT :

بلوک های داده ای تغییر یافته از ناحیه SGA به DATA FILE منتقل و HEADER تمامی DATA BLOCK منتقل شده ، طوری تغییر می کند که وضعیت تایید را نشان دهد و ROLLBACK SEGMENT در حالت غیر استفاده قرار می گیرد تا سایر TRANSACTION ها بتوانند از این فضا برای نگهداری داده های بازگشتی خود استفاده کنند.

• ROLLBACK :

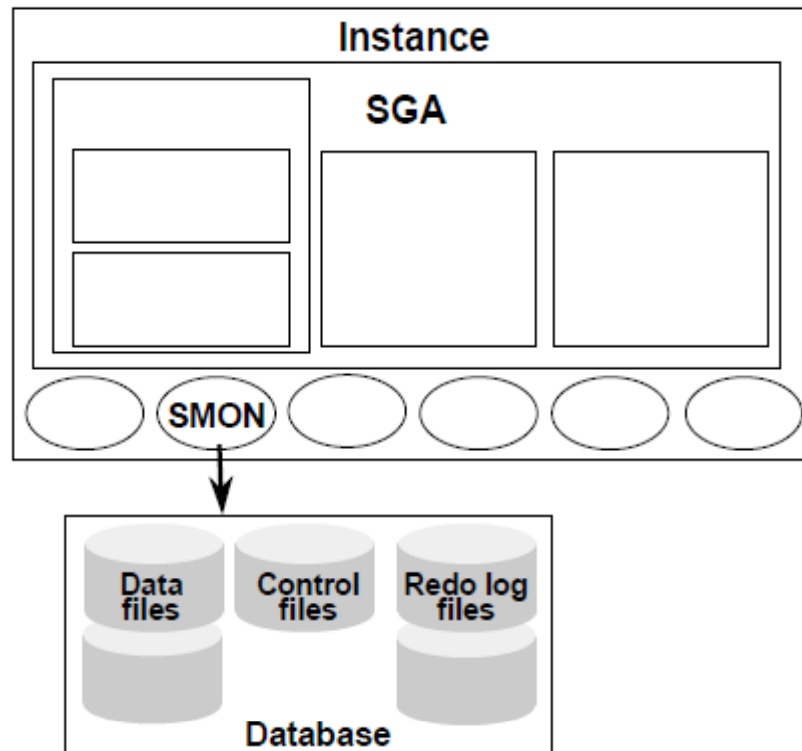
بلوک داده ای تصویر قبلی از ناحیه ROLLBACK SEGMENT به محلی قبلی خود کپی می شود.

و HEADER ، DATA FILE تغییر می کند.

## •Background Process

ارتباط بین ساختارهای فیزیکی وساختارهای حافظه بانک اطلاعاتی اوراکل از طریق این پردازش ها مدیریت می شوند .

### System Monitor (SMON)

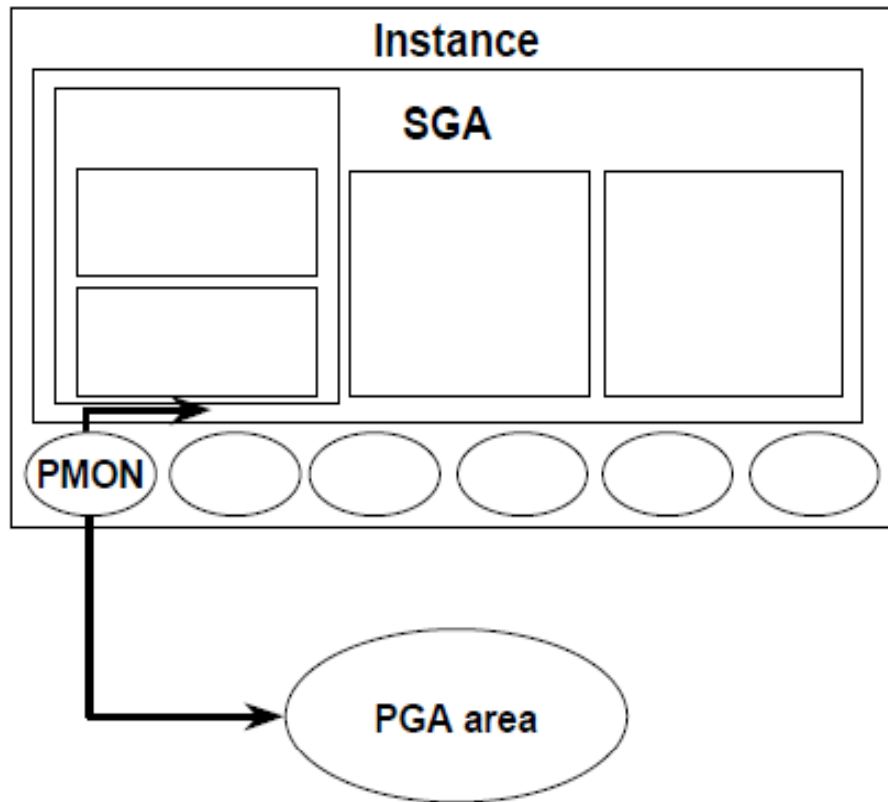


#### Responsibilities:

- Instance recovery:
  - Rolls forward changes in the redo logs
  - Opens the database for user access
  - Rolls back uncommitted transactions
- Coalesces free space ever 3 sec
- Deallocates temporary segments



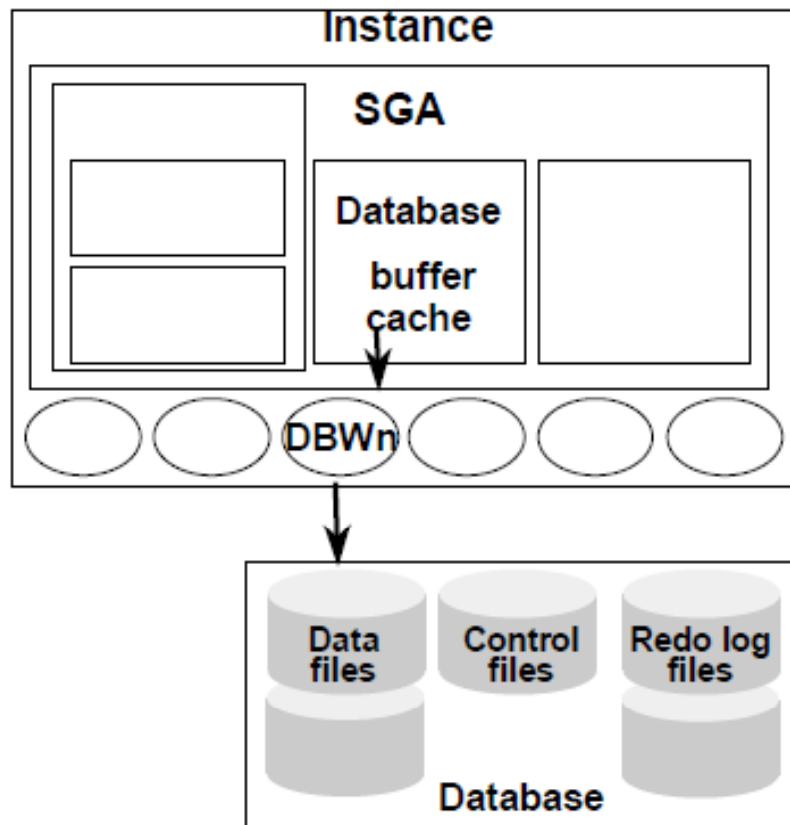
## Process Monitor (PMON)



Cleans up after failed processes by:

- Rolling back the transaction
- Releasing locks
- Releasing other resources
- Restarts dead dispatchers

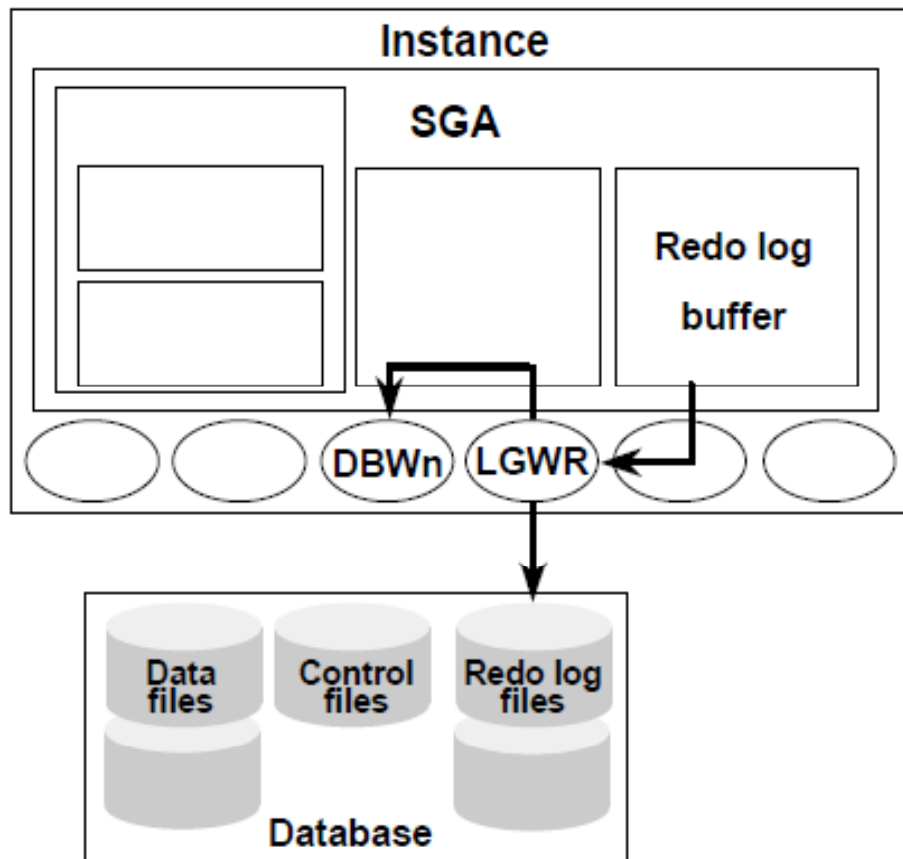
## Database Writer (DBWn)



### DBWn writes when:

- Checkpoint
- Dirty buffers threshold reached
- No free buffers
- Timeout
- RAC ping request
- Tablespace offline
- Tablespace read only
- Table DROP or TRUNCATE
- Tablespace BEGIN BACKUP

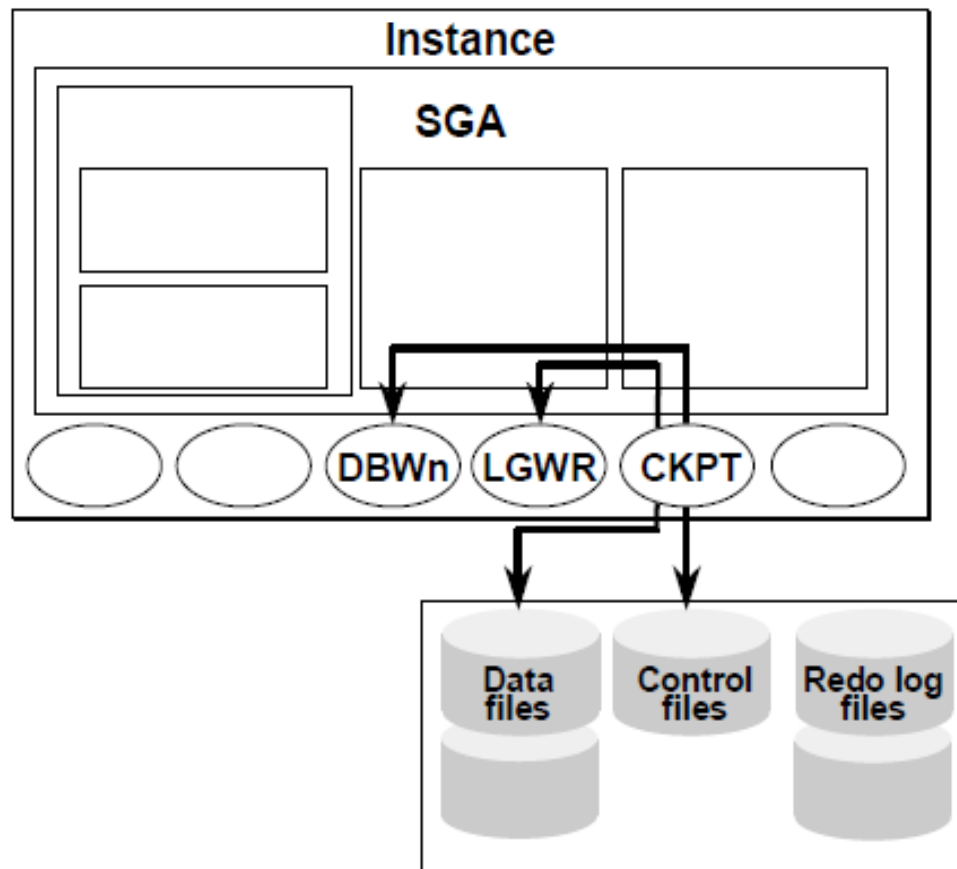
## Log Writer (LGWR)



### LGWR writes:

- At commit
- When one-third full
- When there is 1 MB of redo
- Every 3 seconds
- Before DBWn writes

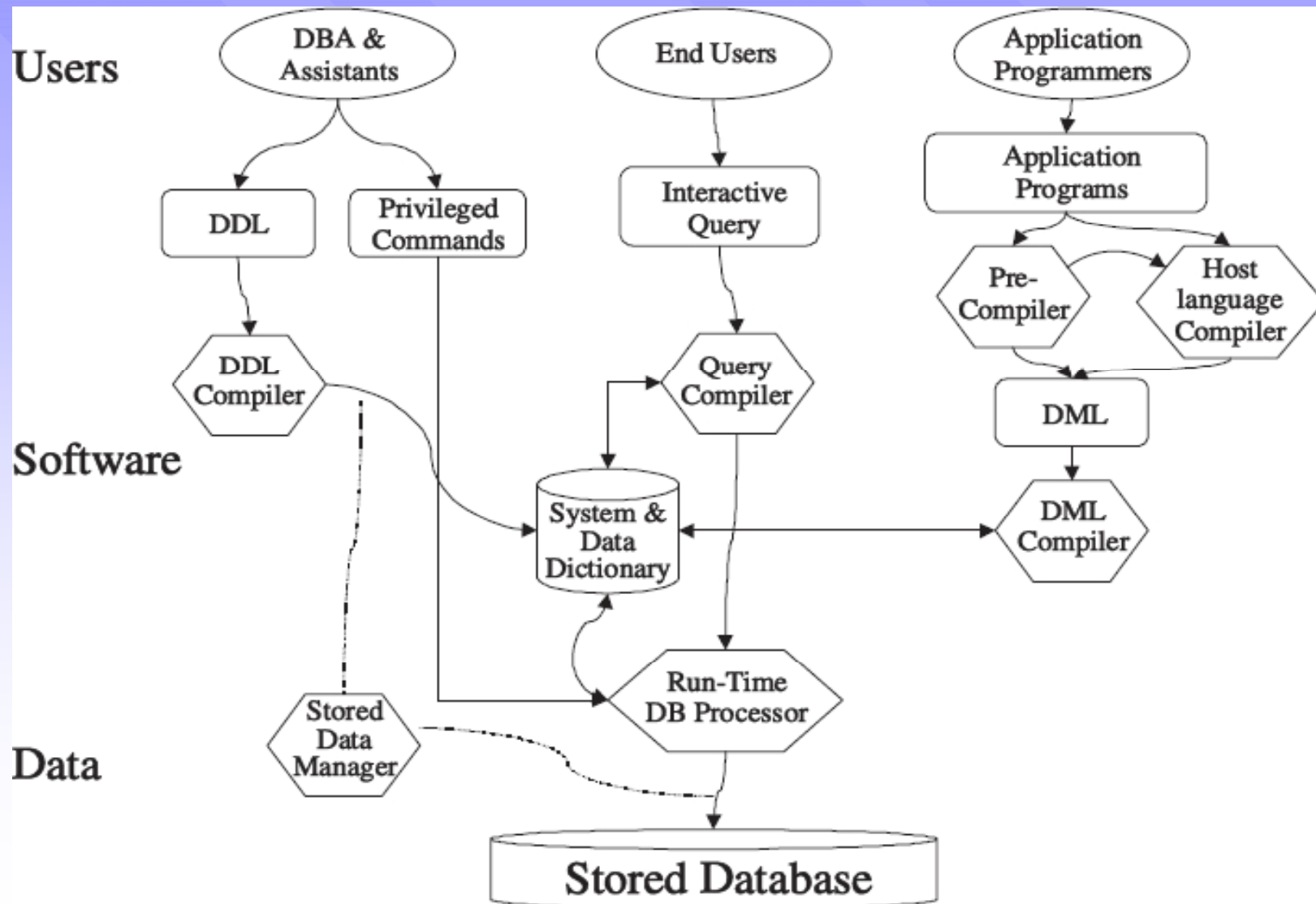
## Checkpoint (CKPT)



Responsible for:

- Signalling DBWn at checkpoints
- Updating datafile headers with checkpoint information
- Updating control files with checkpoint information

**تمرین ۱: الف - مراحل اجرای QUERY در اوراکل**  
**ب - مراحل اجرای دستورات DML در اوراکل**



**جلسه سوم**

**ابزارهای مدیریتی بانک اطلاعاتی**

**و مدیریت INSTANCE**

## Database Administration Tools

Tool	Description
Oracle Universal Installer (OUI)	Used to install, upgrade, or remove software components
Oracle Database Configuration Assistant	A graphical user interface tool that interacts with the OUI, or can be used independently, to create, delete or modify a database
Password File Utility	Utility for creating a database password file
SQL*Plus	A utility to access data in an Oracle database
Oracle Enterprise Manager	A graphical interface used to administer, monitor, and tune one or more databases

# Oracle Database Configuration Assistant

The Oracle Database Configuration Assistant allows you to:

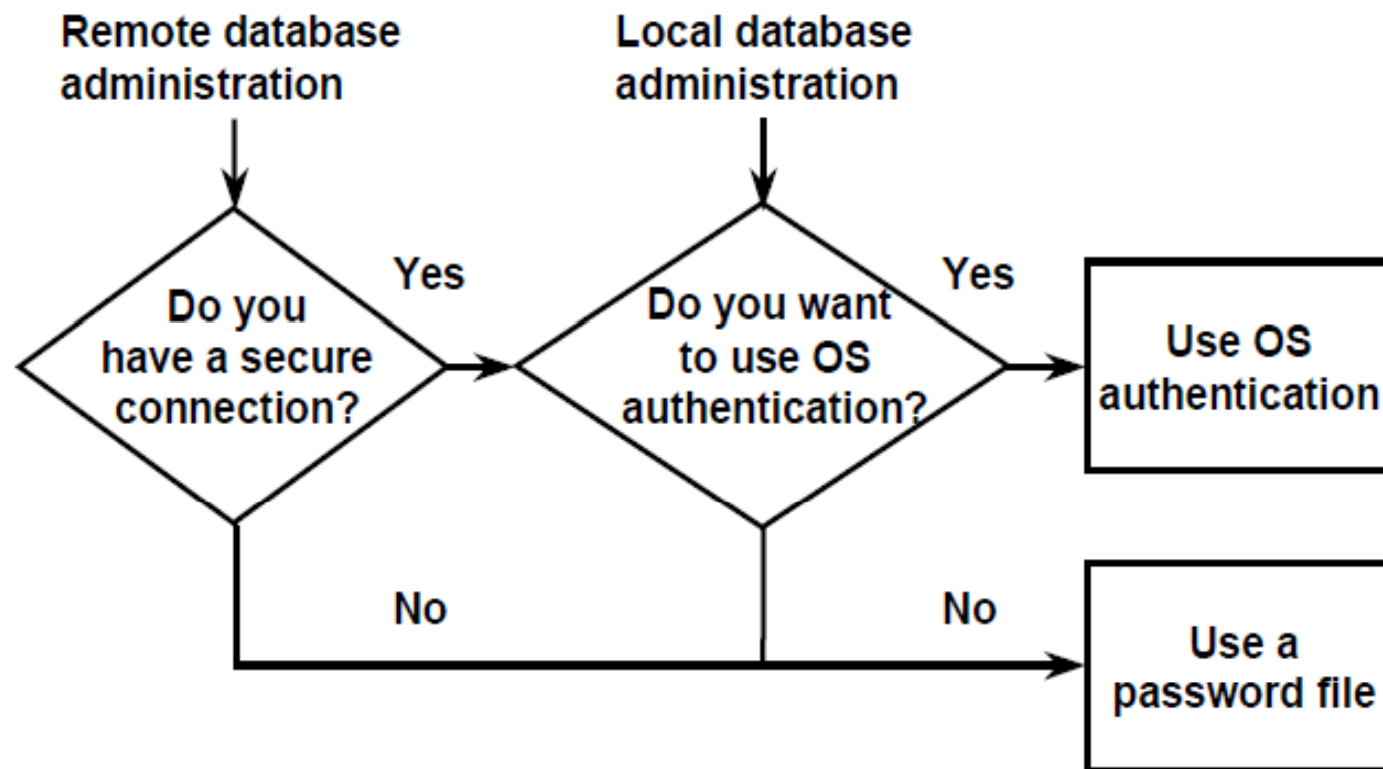
- Create a database
- Configure database options
- Delete a database
- Manage templates



## Oracle Software and File Locations

Software	Files
<code>oracle_base</code>	<code>oradata/</code>
<code>/product</code>	<code>db01/</code>
<code>/release_number</code>	<code>system01.dbf</code>
<code>/bin</code>	<code>control01.ctl</code>
<code>/dbs</code>	<code>redo0101.log</code>
<code>/rdbms</code>	<code>...</code>
<code>/sqlplus</code>	<code>db02/</code>
<code>/admin</code>	<code>system01.dbf</code>
<code>/inst_name</code>	<code>control01.ctl</code>
<code>/pfile</code>	<code>redo0101.log</code>
	<code>...</code>

## Authentication Methods for Database Administrators



## How to Use Password File Authentication (continued)

### Using a Password File

1. Create the password file using the password utility `orapwd`.

```
orapwd file=filename password=password entries=max_users
```

where:

`filename` is the name of the password file

`password` is the password for `SYSOPER` and `SYSDBA`

`entries` is the maximum number of distinct users allowed to connect as `SYSDBA` or `SYSOPER`.

If you exceed this number, you must create a new password file. It is safer to have a larger number.

2. Set the `REMOTE_LOGIN_PASSWORDFILE` parameter to `EXCLUSIVE` or `SHARED`

where:

`EXCLUSIVE` indicates that only one instance can use the password file and that the password file contains names other than `SYS`. Using an `EXCLUSIVE` password file you can grant `SYSDBA` or `SYSOPER` privileges to individual users.

`SHARED` indicates that more than one instance can use the password file. The only user recognized by the password file is `SYS`. You cannot add users to the `SHARED` password file.

3. Connect to the database as follows:

```
CONNECT sys/admin AS SYSDBA
```

## **SQL\*Plus**

**SQL\*Plus** is an Oracle tool that provides the capability to interact with and manipulate the database.

- **Provides the ability to start up and shutdown the database, create and run queries, add rows, modify data, and write customized reports**
- **Subset of the standard SQL language with specific add ons**

### SQL Response Time

.Latest collection is empty !

Edit Baseline

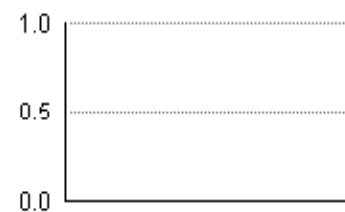
### Active Sessions



No data is currently available.

▲ Maximum CPU

### Host CPU



No data is currently available.

28,40 Paging 1,11 Load

### General

Shutdown



Up Status  
IRST 09:22:56 2-11/11-8 Up Since  
sale Instance Name  
10,2,0,1,0 Version  
Niloufar.Opco.loc Host  
LISTENER\_Niloufar.Opco.loc Listener

[View All Properties](#)

### High Availability

1s (Instance Recovery Time (sec)  
n/a Last Backup  
100 (%) Usable Flash Recovery Area  
Disabled Flashback Logging

Unavailable

•  
1  
1 ✓  
2

### Space Summary

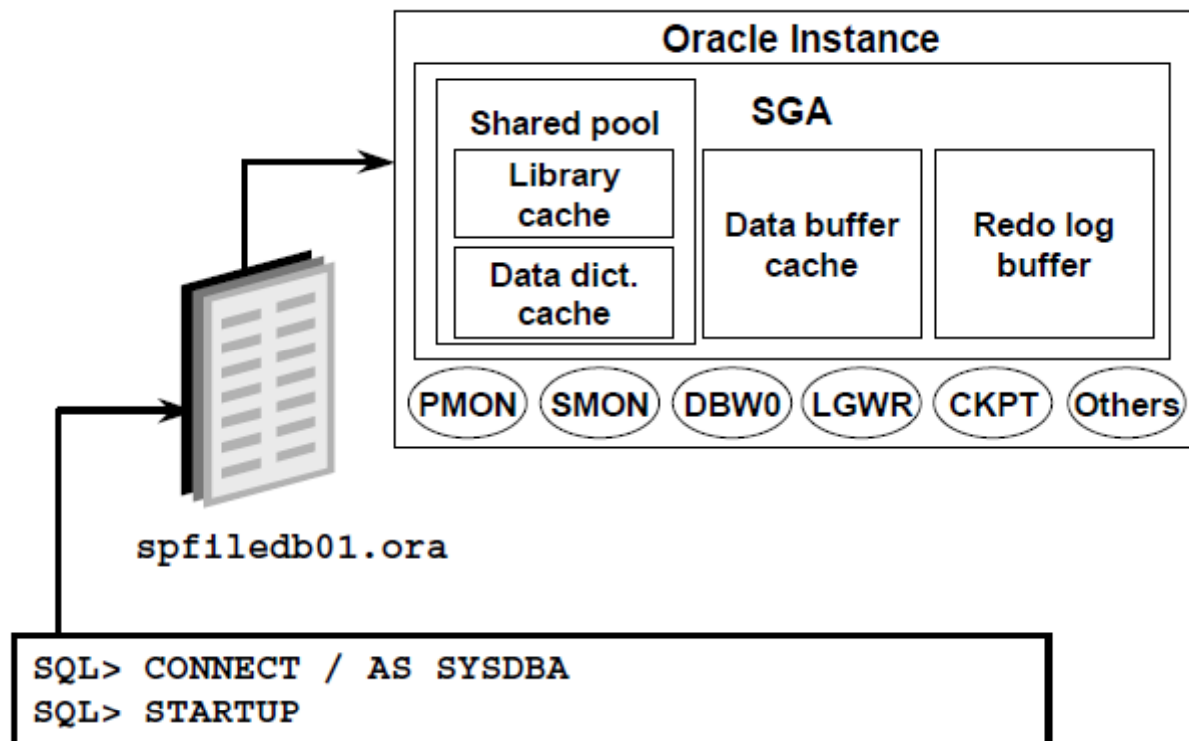
(Database Size (GB)  
Problem Tablespaces  
Segment Advisor  
Recommendations  
Space Violations  
(%) Dump Area Used

### Diagnostic Summary

• ADDM Findings  
47 All Policy Violations  
2,2,13,41 2,11/0,31 Alert Log




















Alerts ▼

## Initialization Parameter Files



## **PFILE** **initSID.ora**

- The **PFILE** is a text file that can be modified with an operating system editor.
- Modifications to the file are made manually.
- Changes to the file take effect on the next startup.
- Its default location is `$ORACLE_HOME/dbs`.

Name	Type	Data
 (Default)	REG_SZ	(value not set)
 MSHELP_TOOLS	REG_SZ	D:\oracle\product\10.2.0\db_1\MSHELP
 NLS_LANG	REG_SZ	AMERICAN_AMERICA.AR8MSWIN1256
 OLEDB	REG_SZ	D:\oracle\product\10.2.0\db_1\oledb\mesg
 OO4O	REG_SZ	D:\oracle\product\10.2.0\db_1\oo4o\mesg
 ORA_SALE_AUTOSTART	REG_EXPAND_SZ	TRUE
 ORA_SALE_SHUTDOWN	REG_EXPAND_SZ	TRUE
 ORA_SALE_SHUTDOWN_TIMEOUT	REG_EXPAND_SZ	90
 ORA_SALE_SHUTDOWNTYPE	REG_EXPAND_SZ	immediate
 ORACLE_BASE	REG_SZ	D:\oracle\product\10.2.0
 ORACLE_BUNDLE_NAME	REG_SZ	Enterprise
 ORACLE_GROUP_NAME	REG_SZ	Oracle - OraDb10g_home1
 ORACLE_HOME	REG_SZ	D:\oracle\product\10.2.0\db_1
 ORACLE_HOME_KEY	REG_SZ	SOFTWARE\ORACLE\KEY_OraDb10g_home1
 ORACLE_HOME_NAME	REG_SZ	OraDb10g_home1
 ORACLE_SID	REG_SZ	SALE
 RDBMS_ARCHIVE	REG_SZ	D:\oracle\product\10.2.0\db_1\DATABASE\ARCHIVE
 RDBMS_CONTROL	REG_SZ	D:\oracle\product\10.2.0\db_1\DATABASE
 SQLPATH	REG_SZ	D:\oracle\product\10.2.0\db_1\dfs





## PFILE Example

```
# Initialization Parameter File: initdb01.ora
db_name                = db01
instance_name          = db01
control_files          = ( /u03/oradata/db01/control01db01.ctl,
                          /u03/oradata/db01/control02db01.ctl)
db_block_size          = 4096
db_block_buffers       = 500
shared_pool_size       = 31457280  # 30M Shared Pool
db_files               = 1024
max_dump_file_size     = 10240
background_dump_dest   = /u05/oracle91/admin/db01/bdump
user_dump_dest         = /u05/oracle91/admin/db01/udump
core_dump_dest         = /u05/oracle91/admin/db01/cdump
undo_management        = auto
undo_tablespace        = undtbs
. . .
```

## SPFILE spfileSID.ora

- Binary file with the ability to make changes persistent across shutdown and startup
- Maintained by the Oracle server
- Records parameter value changes made with the `ALTER SYSTEM` command
- Can specify whether the change being made is temporary or persistent
- Values can be deleted or reset to allow an instance to revert to the default value

```
ALTER SYSTEM SET undo_tablespace = 'UNDO2';
```

## Creating an SPFILE

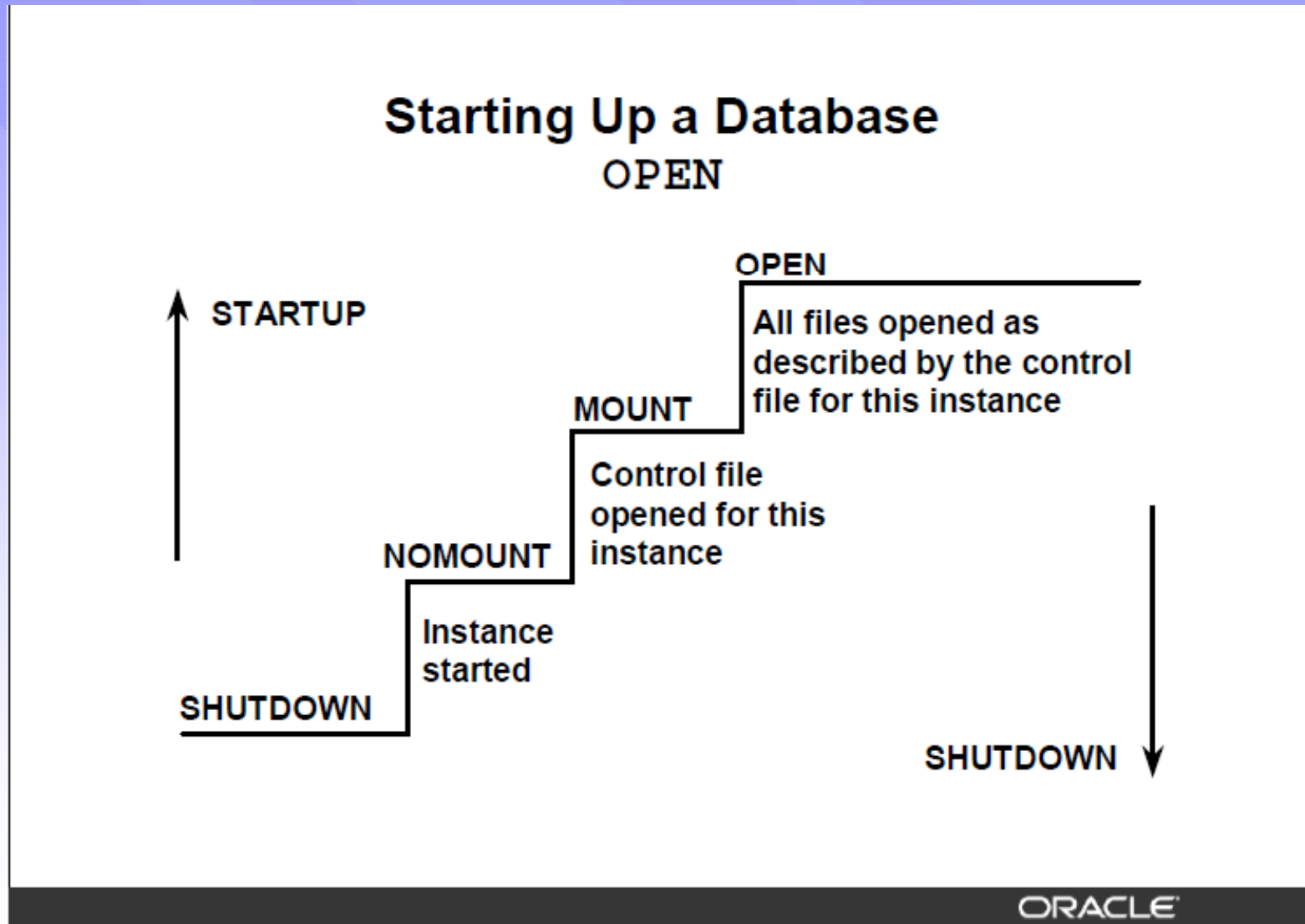
SPFILE can be created from an `initSID.ora` file using the `CREATE SPFILE` command, which can be executed before or after instance startup:

```
CREATE SPFILE FROM PFILE;
```

## SPFILE Example

```
*.background_dump_dest='$ORACLE_HOME/admin/db01/bdump'
*.compatible='9.0.0'
*.control_files='/u03/oradata/db01/ctrl01db01.ctl','/u03/orad
ata/db01/ctrl02db01.ctl'
*.core_dump_dest='$ORACLE_HOME/admin/db01/cdump'
*.db_block_buffers=500
*.db_block_size=4096
*.db_files=40
*.db_name='db01'
*.instance_name='db01'
*.remote_login_passwordfile='exclusive'
*.shared_pool_size=31457280 # 30M Shared Pool
*.undo_management='AUTO'
db01.undo_tablespace='UNDOTBS01'
db02.undo_tablespace='UNDOTBS02'
. . .
```

## INSTACCE بانک اطلاعاتی می تواند شامل حالت های زیر باشد:



## الف) SHUTDOWN :

در این حالت INSTANCE که مجموعه ای از پردازشها و ساختار حافظه برای دستیابی به داده های بانک اطلاعاتی است وجود نداشته و بانک اطلاعاتی در دسترس کاربر قرار ندارد. در این حالت تنها کاربرانی که دارای مجوز SYSOPR , SYSDBA می باشند می توانند به بانک اطلاعاتی وصل شده و آنرا در حالت START ویا مورد RECOVERY قرار دهند. SHUTDOWN بانک اطلاعاتی به حالت زیر امکان پذیر است :

### Shutting Down the Database

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

#### Shutdown Mode:

- NORMAL
- TRANSACTIONAL
- IMMEDIATE
- ABORT

x	NO
o	YES

❑ NORMAL : امکان ایجاد SESSION جدید وجود نداشته و زمانی که تمامی کاربران به دلخواه خود از بانک اطلاعاتی خارج می شوند بانک را در حالت SHUTDOWN قرار می دهد.

❑ IMMEDIATE : بلا فاصله ارتباط تمامی کاربران را از بانک اطلاعاتی قطع کرده و تمامی TRANSACTION های ثبت نشده را ROLLBACK می کند .

❑ TRANSACTIONAL : بلا فاصله ارتباط تمامی کاربران را از بانک اطلاعاتی قطع کرده و تمامی TRANSACTION های ثبت نشده را COMMIT می کند .

❑ ABORT : همانند قطع برق باعث از بین رفتن INSTANCE بانک اطلاعاتی می گردد باید

توجه داشت که در STARTUP بعدی نیاز به RECOVERY بانک اطلاعاتی وجود دارد .

## مراحل زیر در زمان SHUTDOWN صورت می گیرد :

### ۱- CLOSE :

ثبت آخرین تغییرات بانک اطلاعاتی در DATA FILE ها و REDO LOG FILE های بانک اطلاعاتی و خارج کردن این ساختار فیزیکی از INSTANCE بانک. بنابراین در زمان SHUTDOWN بانک اطلاعاتی عمل CHECKPOINT انجام می شود یعنی :

- ❑ ذخیره تغییرات بانک اطلاعاتی از REDO LOG BUFFER به REDO LOG FILE ها
- ❑ ثبت بلوک های دادهای تغییر یافته از ناحیه DATA BLOCK BUFFER به DATA FILE ها توسط DBRW
- ❑ ثبت آخرین شماره تغییر بانک اطلاعاتی در CONTROL FILE

### ۲- DOSMOUNT :

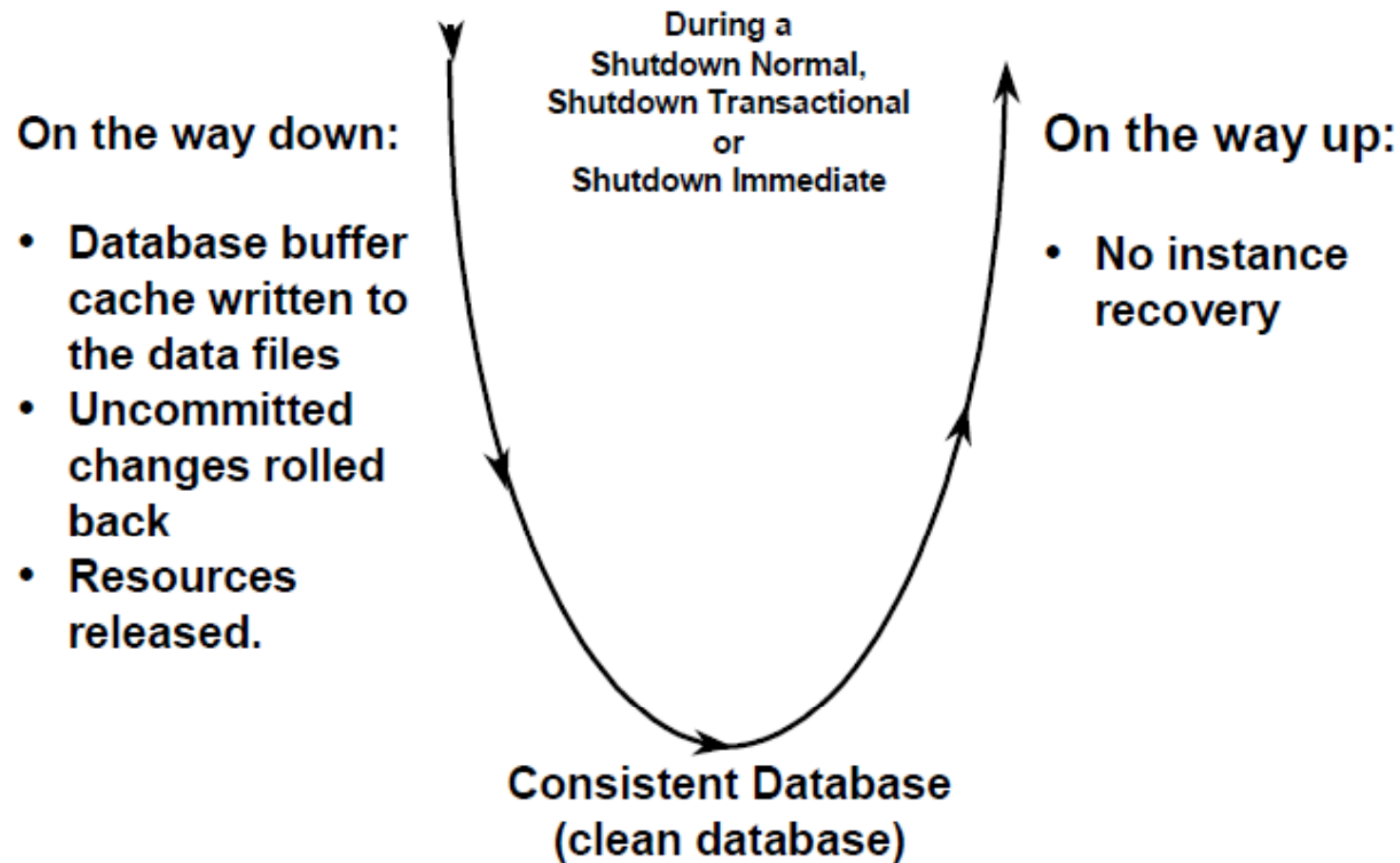
ثبت آخرین وضعیت بانک اطلاعاتی در CONTROL FILE ها و خارج کردن این فایل های فیزیکی از INSTANCE بانک اطلاعاتی

### ۳- ORACLE INSTANCE SHUTDOWN :

در این مرحله ابتدا پردازشهای INSTANCE از بین می روند و سپس ساختارهای حافظه ای INSTANCE شامل نواحی PGA, SGA آزاد شده و به سستم عامل باز می گردد.



# Shutdown Options



## Shutdown Options

### On the way down:

- Modified buffers are not written to the data files
- Uncommitted changes are not rolled back

During a  
Shutdown Abort  
or  
Instance Failure  
or  
Startup Force

### On the way up:

- Redo logs used to reapply changes
- Undo segments used to roll back uncommitted changes
- Resources released

Inconsistent Database  
(dirty database)

## برای START شدن بانک اطلاعاتی به سه مرحله زیر نیاز می باشد:

OPEN\*

MOUNT\*

NOMOUNT\*

ب ( NOMOUNT :

در این مرحله جستجو برای وجود فایل پارامتری INIT.ORA یا SPFILE[SID].ORA انجام شده و پارامترهای زیر از آن خوانده می شود:

□ نام بانک اطلاعاتی

□ نام و مسیر CONTROL FILE های بانک اطلاعاتی

□ نام INSTANCE بانک اطلاعاتی

□ VERSION بانک اطلاعاتی

□ نام UNDO TABLESPACE که وظیفه نگهداری داده های بازگشتی را دارد

□ و..... سایر پارامترهای INSTANCE

## ج) MOUNT :

در این مرحله وجود یا عدم وجود CONTROL FILE های بانک اطلاعاتی در مسیری که قبلاً در مرحله NOMOUNT از فایل پارامتری INSTANCE خوانده است، انجام می شود و اگر CONTROL FILE ها وجود داشته باشند آنگاه اطلاعات زیر خوانده میشود :

❑ نام بانک اطلاعاتی

❑ ID یکتای بانک اطلاعاتی : در زمان ایجاد بانک اطلاعاتی به DB تخصیص تخصیص داده می شود و در HEADER تمامی DATA FILE و REDO LOG FILE ها ثبت می شود و به معنای آن است که آیا آن فایل عضوی از DB است یا نه ؟

❑ نام و مسیر DATA FILE های بانک اطلاعاتی

❑ نام و مسیر REDO LOG FILE های بانک اطلاعاتی

❑ نام REDO LOG FILE جاری : به LOG FILE ای گفته می شود که در INSTANCE قبلی تغییرات بانک در آن ثبت می شود.

❑ آخرین شماره تغییر بانک اطلاعاتی (SYSTEM CHANGE NUMBER)

❑ اطلاعات مربوط به TABLESPACE های بانک اطلاعاتی

## د ( OPEN :

در این مرحله وجود یا عدم وجود DATA FILE , REDO LOG FILE های بانک اطلاعاتی بررسی می شود. اگر تمامی این فایل ها در دسترس INSTANCE بانک اطلاعاتی قرار داشته باشند آنگاه کنترل های زیر روی این فایل ها اعمال می شود :

□ نام بانک اطلاعاتی که در DATA FILE، HEADER ها نوشته شده است با نام مشخص شده CONTROL FILE مطابقت داده می شود.

□ ID یکتای آن فایل با ID یکتای مشخص شده در CONTROL FILE مطابقت داده می شود.

□ شماره تغییر بانک اطلاعاتی SCN در HEADER آن DATA FILE با شماره SCN در CONTROL FILE مطابقت داده می شود.

## Starting Up

To start up an instance, use the following command:

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

**Note:** This is not the complete syntax.

where:

OPEN	enables users to access the database
MOUNT	mounts the database for certain DBA activities but does not provide user access to the database
NOMOUNT	creates the SGA and starts up the background processes but does not provide access to the database
PFILE= <i>parfile</i>	enables a nondefault parameter file to be used to configure the instance



## The ALTER DATABASE Command

- Change the state of the database from NOMOUNT to MOUNT:

```
ALTER DATABASE db01 MOUNT;
```

- Open the database as a read-only database:

```
ALTER DATABASE db01 OPEN READ ONLY;
```

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

To prevent data from being modified by user transactions, the database can be opened in read-only mode.

To start up an instance, use the following command:

```
ALTER DATABASE OPEN [READ WRITE | READ ONLY]
```

where:

READ WRITE	opens the database in read-write mode, so that users can generate redo logs
------------	---

READ ONLY	restricts users to read-only transactions, preventing them from generating redo log information
-----------	---



## Opening a Database in Restricted Mode

- 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;
```

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

where:

ENABLE RESTRICTED SESSION: enables future logins only for users who have the RESTRICTED SESSION privilege

DISABLE RESTRICTED SESSION: disables RESTRICTED SESSION so that users who do not have the privilege can log on

## Terminate Sessions

After placing an instance in restricted mode, you may want to kill all current user sessions before performing administrative tasks. This can be done by the following:

```
ALTER SYSTEM KILL SESSION 'integer1,integer2'
```

where:

integer1: is the value of the SID column in the V\$SESSION view

integer2: is the value of the SERIAL# column in the V\$SESSION view

جلسہ چہارم

# Backup & Recovery

## سیاست های ترمیم ، پشتیبانی و بارگذاری مجدد

### انواع Backup:

سه روش برای Backup از بانک اطلاعاتی ORACLE وجود دارد:

۱- Export

۲- Backup های Offline

۳- Backup های Online (ARCHIVELOG)

هر Export یک Backup منطقی از بانک اطلاعاتی است و Online Backup و Offline

Backup به عنوان Backup فیزیکی بانک اطلاعاتی می باشند.

Backup های بانک اطلاعاتی اوراکل به دو گروه زیر تقسیم می شوند:

## الف) Backup های منطقی

در این نوع Backup به جای کپی از ساختارهای فیزیکی از محتوی درون بلوکهای داده ای برای انجام Backup استفاده می شود. بنابراین تعاریف ایجاد تمامی اشیا بانک اطلاعاتی در این نوع Backup به صورت دستورات DDL (Data Definition Language) ذخیره می شوند. این نوع Backup تنها می تواند از محتوی بلوکهای داده ای Data file های بانک Backup بگیرد. و در واقع نمی توان از Control File ها و Redo log file برای Backup های منطقی استفاده کرد.

## ب) Backup های فیزیکی

به Backup ای گفته می شود که از ساختارهای فیزیکی بانک اطلاعاتی مانند Data file و Redo log File، Archive log File ها و Control File ها انجام می شود. این نوع Backup ها به دو گروه تقسیم می شوند:

## ۱ – Cold Backup

به Backup ای گفته می شود که در زمان shutdown بودن بانک اطلاعاتی از ساختارهای فیزیکی آن انجام می شود. در این نوع Backup به علت قرار داشتن بانک اطلاعاتی در حالت shutdown امکان تغییر محتوی بلوکهای داده ای Data file های بانک اطلاعاتی وجود نداشته بنابراین می توان با استفاده از فرمان کپی سیستم عامل از ساختارهای فیزیکی بانک Backup تهیه کرد. بانک اطلاعاتی می تواند به حالت های Immediate, Normal و یا Transactional در حالت shutdown قرار گیرد.

قرار دادن بانک اطلاعاتی به هر کدام از حالت های فوق در وضعیت shutdown باعث می شود که تمامی تغییرات اعمال شده با آخرین وضعیت ممکن ثبت شوند. در واقع هیچ تغییری در بانک اطلاعاتی وجود ندارد که وضعیت آن مشخص نشده باشد. بنابراین در صورت نیاز به Restore کردن فایل های Backup به Media Recovery نیاز نمی باشد اما اگر زمان زیادی از انجام Backup گذشته باشد نیاز به Media Recovery بعد از Restore کردن فایل های backup وجود دارد.

به Backup ای گفته می شود که در زمان start بودن بانک اطلاعاتی از ساختارهای فیزیکی آن انجام می شود. در این نوع Backup به علت Open بودن بانک اطلاعاتی امکان تغییر محتوی بلوکهای داده ای بانک توسط کاربران وجود دارد. بنابراین Hot Backup تنها رکوردهای Commit شده در بلوکهای داده ای را در نظر می گیرد. برای انجام Hot Backup در یک بانک اطلاعاتی نیاز به قرار دادن بانک در حالت Archive Log Mode می باشد. بنابراین اگر بانک اطلاعاتی در حالت Archive Log mode قرار نداشته باشد تنها انجام Cold Backup از بانک اطلاعاتی امکان پذیر است و امکان انجام Hot Backup وجود ندارد.

مراحل انجام Hot Backup به صورت زیر است:

۱- قرار دادن Tablespace در حالت Backup با دستور:

```
ALTER TABLESPACE tbs_name BEGIN BACKUP;
```

۲- انجام کپی از Data file های آن Tablespace با فرمان کپی سیستم عامل

۳- خارج کردن Tablespace از حالت Backup با دستور:

```
ALTER TABLESPACE tbs_name END BACKUP;
```

Example: EXP SCOTT/TIGER

Or, you can control how Export runs by entering the EXP command followed by various arguments. To specify parameters, you use keywords:

Format: EXP KEYWORD=value or KEYWORD=(value1,value2,...,valueN)

Example: EXP SCOTT/TIGER GRANTS=Y TABLES=(EMP,DEPT,MGR)  
or TABLES=(T1:P1,T1:P2), if T1 is partitioned table

USERID must be the first parameter on the command line.

Keyword	Description (Default)	Keyword	Description (Default)
USERID	username/password	FULL	export entire file (N)
BUFFER	size of data buffer	OWNER	list of owner usernames
FILE	output files (EXPDAT.DMP)	TABLES	list of table names
COMPRESS	import into one extent (Y)	RECORDLENGTH	length of IO record
GRANTS	export grants (Y)	INCTYPE	incremental export type
INDEXES	export indexes (Y)	RECORD	track incr. export (Y)
DIRECT	direct path (N)	TRIGGERS	export triggers (Y)
LOG	log file of screen output	STATISTICS	analyze objects (ESTIMATE)
ROWS	export data rows (Y)	PARFILE	parameter filename
CONSISTENT	cross-table consistency(N)	CONSTRAINTS	export constraints (Y)
OBJECT_CONSISTENT	transaction set to read only during object export (N)		
FEEDBACK	display progress every x rows (0)		
FILESIZE	maximum size of each dump file		
FLASHBACK_SCN	SCN used to set session snapshot back to		
FLASHBACK_TIME	time used to get the SCN closest to the specified time		
QUERY	select clause used to export a subset of a table		
RESUMABLE	suspend when a space related error is encountered(N)		
RESUMABLE_NAME	text string used to identify resumable statement		
RESUMABLE_TIMEOUT	wait time for RESUMABLE		
TTS_FULL_CHECK	perform full or partial dependency check for TTS		
TABLESPACES	list of tablespaces to export		
TRANSPORT_TABLESPACE	export transportable tablespace metadata (N)		
TEMPLATE	template name which invokes iAS mode export		

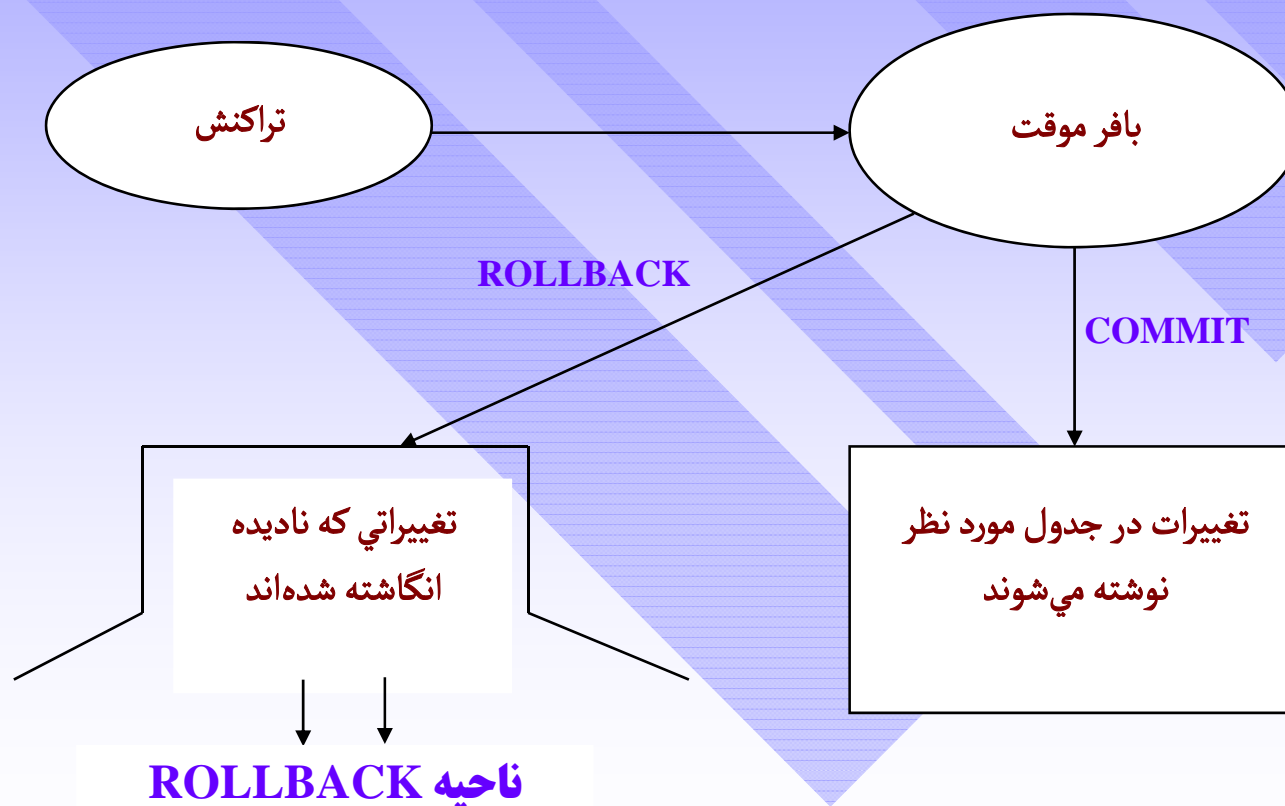


جلسہ پنجم

**TRANSACTION**

# تراکنش (Transaction):

هر برنامه ای که توسط کاربر در محیط بانک اطلاعاتی اجرا میشود تراکنش نام دارد. تراکنش یک واحد منطقی از کار است و معمولاً شامل چندین عمل بانک اطلاعاتی است.



## تفاوت اصلی یک تراکنش با یک برنامه معمولی :

تفاوت اصلی یک تراکنش با یک برنامه معمولی در محیط غیر بانکی این است که تراکنش همواره به DBMS تسلیم می شود و DBMS در اعمال هرگونه کنترل و حتی به تعویق انداختن و ساقط کردن آن آزادی عمل دارد.

## هدف اصلی تراکنش ها :

هدف اصلی این کنترل ها حفظ جامعیت و صحت بانک اطلاعاتی است . چرا که در بانک اطلاعاتی آنچه در درجه اهمیت دارد داده است نه برنامه . داده های بانک اطلاعاتی را مانا (PERSISTENT) می نامند برنامه ها می آیند و میروند اما داده ها می مانند . مثلاً برنامه ای که پولی را به حسابی می ریزد یا برداشت میکند آنقدر مهم نیست . مهم این است که موجودی حسابها اشتباه نشود.

## تضمین جامعیت بانک اطلاعاتی :

آقای جیم گری (Jim Gary) در سال ۱۹۸۱ ثابت کرد که چهار کنترل زیر لازم است روی تمامی تراکنش ها در بانک اطلاعات اعمال گردد تا صحت و جامعیت آن تضمین شود این کنترل ها به خواص ACID معروفند.

۱- یکپارچگی (Atomicity)

۲- همخوانی (Consistency)

۳- انزوا (Isolation)

۴- پایداری (Durability)

## ۱- یکپارچگی (Atomicity):

این خاصیت به همه یا هیچ موسوم است. منظور این است که یا تمام دستورات تراکنش باید اجرا شود یا هیچکدام از آنها نباید اجرا شود. مثلاً تراکنشی می‌خواهد مبلغی را از حسابی به حساب دیگر منتقل کند. فرض کنید بخش اول کار برداشت پول در یک ماشین و بخش دوم کار واریز پول در ماشینی دیگر انجام شود. حال در نظر بگیرید پس از انجام بخش اول برداشت پول ارتباط با ماشین دوم ناگهان قطع شود بدیهی است که در این حالت باید پول برداشت شده به همان حساب بازگردانده شود.

## ۲- همخوانی (Consistency):

این خاصیت به این صورت بیان می‌گردد که: (هر تراکنشی اگر به تنهایی اجرا شود بانک اطلاعات را از حالتی صحیح به حالت صحیح دیگری منتقل می‌کند) یعنی خاصیت می‌گوید که هر تراکنش باید تمامی قوانین جامعیت بانک اطلاعاتی را رعایت کند. تراکنش ممکن است دو نوع پایان داشته باشد :

الف) پایان نا موفق که آن را سقوط (Rollback) می‌نامند.

ب) پایان موفق که آن را انجام (Commit) می‌نامند.

### ۳- انزوا (Isolation) :

در بانک اطلاعاتی ممکن است تراکنش های همروند وجود داشته باشد . بر طبق خاصیت انزوا ، همروندی تراکنش ها باید کنترل شود تا اثر مخرب بر روی هم نداشته باشند به عبارت دیگر اثر تراکنش های همروند، روی یکدیگر چنان است که گویا هر کدام در انزوا انجام می شود. یعنی به هنگام سازی T1 توسط تراکنش دیگری مثل T2 قابل مشاهده نیست مگر اینکه T1 عمل COMMIT را اجرا کند.

### ۴- پایداری (Durability) :

بر اساس این خاصیت تراکنش هایی که به مرحله انجام (COMMIT) برسند اثرشان ماندنی است و هرگز به طور تصادفی از بین نمی روند . مثلا اگر مبلغی به حسابی واریز شود تراکنش مربوطه انجام یافته، حتی در صورت وقوع آتش سوزی در آن شعبه بانک، مشتری نباید متضرر شود یعنی عمل واریز قبل از اعلام انجام موفق باید در جای دیگری نیز ثبت شده باشد.

## استقلال داده ها (Data Independence):

منظور از استقلال داده ها مستقل بودن ذخیره سازی داده ها از کاربرد آنها است و به دو صورت تعبیر می شود:

### ۱- استقلال فیزیکی داده ها (Physical Data Independence):

در استقلال فیزیکی داده ها اگر تغییری در ذخیره سازی داده ها انجام گیرد (مثلا نوع دیسک عوض شود) برنامه های کاربردی هیچ تغییری نمی کنند .

### ۲- استقلال منطقی داده ها (Logical Data Independence):

در استقلال منطقی داده ها تغییر تصویر ادراکی بانک از دید کاربران و برنامه های آنها مخفی می ماند. مثلا اگر جدولی چهار ستون داشته و برنامه هایی روی آن ستونها نوشته شده در صورتی که ستون پنجمی به آن اضافه شود برنامه های سابق نیاز به دستکاری ندارند و با همان شکل قبلی قابل اجرا هستند .

### نکته مهم :

نگاشت مفهومی / داخلی (در معماری ANSI) کلید استقلال فیزیکی داده هاست. و نگاشتها خارجی/مفهومی کلید استقلال منطقی داده هاست. به عبارت دیگر در استقلال منطقی طراح در تغییرات سطوح ادراکی/ خارجی آزاد است و این تغییرات شامل موجودیتها صفات خاصه و ارتباط بین موجودیتهاست. در استقلال فیزیکی طراح در تغییرات سطوح داخلی /ادراکی آزاد است و این تغییرات شامل ساختار فایلها ساختارهای ذخیره سازی<sup>۷</sup> دستگاههای ذخیره سازی متفاوت است.

# Trigger

انواع متفاوتی از قیود جامعیت که تاکنون بر روی جداول بحث شده اند اجزای اعلانی ساده ای از شروط را فراهم می کردند که بر روی جداول تعریف می شدند که عبارت بودند از کلید اصلی و کلید خارجی و قیود حوزه .

قیود جامعیت پیچیده که به چندین صفت یا جدول اشاره می کنند نمی توانند در داخل تعریف جدول مشخص شوند . در عوض تریگرها یک تکنیک رویه ای را جهت نگهداری و مشخص کردن قیود جامعیت را فراهم می کنند . حتی تریگرها به کاربران اجازه مشخص کردن قیود جامعیت پیچیده را می دهند برای اینکه تریگرها اصولاً رویه های PL/SQL هستند . مانند یک رویه به یک جدول وابسته هستند و بطور خودکار توسط پایگاه داده فراخوانی می شوند هر گاه تغییراتی در جدول صورت بگیرد . تغییرات بر روی جدول ممکن است شامل delete , update , insert باشد .



## ساختار تریگرها

اجزای تعریف کننده تریگر شامل اجزای زیر است :

• نام تریگر

create [ or replace ] trigger < trigger name >

• trigger time point before | after

• رخداد های تریگرها

insert or update [ of < column(s) > ] or delete on < table >

• نوع تریگر ( اختیاری )

برای هر سطر

• محدودیت تریگرها ( تریگرها برای هر سطر )

when ( < condition > )

• بدنه تریگر < PL/SQL block >

بند replace یک تریگر یکسانی را که ایجاد شده بود را دوباره خلق می کند . نام تریگر می تواند بصورت

دلخواه انتخاب شود اما برنامه نویسی خوب است که نام تریگر بصورت upd\_ins\_EMP باشد یعنی هم

اعمالی که بر روی جدول انجام می شود و هم نام جدول در نام تریگر بیاید.

## لغت نامه داده ها و کاتالوگ سیستم :

لغت نامه داده ها (Data Dictionary) شبیه لغتنامه های معمولی تمامی اسامی استفاده شده در سیستم و معنای آنها را در بر می گیرد . این اسامی شامل تمامی نامهای جدول شی ها صفت ها و غیره است .

علاوه بر اسامی داده ها اطلاعات دیگری باید در مورد بانک نگهداری شود مثل اطلاعات مربوط به حق دسترسی افراد به دادهای مختلف ، تاریخ ایجاد ویا تغییر داده ها تعداد نسخه های هر پرونده اندازه هر جدول یا شی و غیره . اینگونه اطلاعات در کاتالوگ سیستم (System Catalog) نگهداری میشود. در واقع لغتنامه داده ها زیر مجموعه کاتالوگ سیستم است ولی بدلیل کاربرد ویژه آن مجزا شده و برای کار با آن نرم افزار خاصی طراحی شده است. سیستم DBMS به طور خود کار و به کمک کاربران اطلاعات موجود در کاتالوگ سیستم را همواره به روز نگه می دارد.

## مطالبی که در کاتالوگ سیستم ذخیره می شوند عبارت اند از :

- نام ساختار داده ها ، مثلا نام جدول ها در مدل بانک رابطه ای
- نام موجودیت ها و اتباطات بین آنها
- نام صفات خاصه هر نوع موجودیت ، نوع و محدوده مقادیر آنها
- شماهای خارجی و ادراکی و رویه های تبدیل بین سطوح مختلف و نیز شماای داخلی
- مشخصات کاربران و چگونگی حق دستیابی آنها به داده ها و محدوده مجاز عملیات آنها
- مشخصات سیستمی پایانه های متصل به بانک
- تاریخ ایجاد داده ها، مکانیسم ورود داده ها به بانک و چگونگی استفاده از آنها

## جداول : Data Dictionary

نصب بانک اطلاعاتی اراکل همیشه شامل ایجاد سه کاربر استاندارد اراکل می باشد :

**Sys** : مالک همه جداول و view های ، data dictionary می باشد . این کاربر دارای بالاترین حق

امتياز برای مدیریت اشیاء و ساختار بانک اطلاعاتی اراکل مانند ایجاد کاربران جدید است .

**System** : مالک جداولی است که توسط ابزار مختلف مانند اشکال \* SQL و SQL \* Report و

...مورد استفاده قرار می گیرند . این کاربر دارای حق دسترسی در سطح پایین تری نسبت به sys

است .

**Public** : یک کاربر مصنوعی در بانک اطلاعاتی اراکل می باشد . همه امتیازات بصورت خودکار

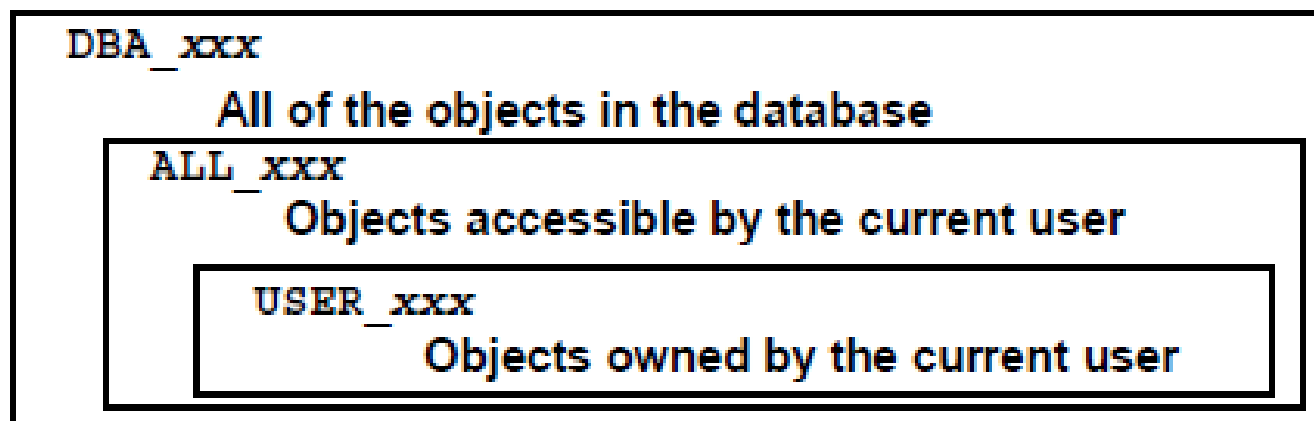
برای این کاربر در نظر گرفته شده و همه کاربران بانک اطلاعاتی آنرا می شناسند .

# :Data Dictionary Views

## Data Dictionary View Categories

The data dictionary consists of three main sets of static views distinguished from each other by their scope:

- **DBA:** What is in all the schemas
- **ALL:** What the user can access
- **USER:** What is in the user's schema



**USER\_** : تاپل ها در view ، USER شامل اطلاعاتی درباره اشياء شخصی ( کاربر فعلی )

**USER\_TABLES** : همه جداول با نام و تعداد ستون هایشان و اطلاعاتی در مورد ذخیره سازی آنها

**USER\_CATALOG** : جداول ، view ها و synonym ها ( CAT )

**USER\_COL\_COMMENTS** : توضیحات بر روی ستون ها

**USER\_CONSTRAINTS** : تعریف قيود برای جداول

**USER INDEXES D** : تمامی اطلاعات در مورد خلق شاخص ها بر روی جداول ( IND )

**USER\_OBJECTS** : تمامی اشياء بانک اطلاعاتی کاربر فعلی ( OBJ )

**USER\_TAB\_COLUMNS** : ستون های جداول و view های شخصی کاربر ( COLS )

**USER\_TAB\_COMMENTS** : توضیحات بر روی جداول و view ها

**USER\_TRIGGERS** : تریگرهای تعریف شده توسط کاربران

**USER\_USERS** : اطلاعاتی درباره کاربر فعلی **USER\_VIEWS** : view های تعریف شده کاربران

**ALL\_** : سطر ها در view های ALL شامل سطرهایی از view های user و اطلاعاتی درباره اشیا که برای کاربر فعلی قابل دسترس هستند . ساختار این view ها مشابه ساختار view های USER است .  
**ALL\_CATALOG** : نام و نوع تمامی جداول در دسترس و view ها و synonym ها .

**ALL\_TABLES** : نام جداول شخصی در دسترس

**ALL\_OBJECTS** : نوع و نام اشیا بانک اطلاعاتی در دسترس

**ALL\_TRIGGERS** : .....

**ALL\_USERS** : .....

**ALL\_VIEWS** : .....

**DBA\_** : view های DBA شامل اطلاعاتی درباره اشیا بانک اطلاعاتی صرف نظر از view های شخصی می باشد .

**DBA\_TABLES** : جداول تمامی کاربران در بانک اطلاعاتی

**DBA\_CATALOG** : جداول و view ها و synonym های تعریف شده بر روی جداول .

**DBA\_OBJECTS** : اشیا تمامی کاربران

**DBA\_DATA\_FILES** : اطلاعاتی در مورد فایل های داده

**DBA\_USERS** : اطلاعاتی درباره تمامی کاربرانی که بانک اطلاعاتی آنها را می شناسد .

## Systemwide Statistics

<u>Instance/Database</u>	
V\$DATABASE	T
V\$INSTANCE	T
V\$OPTION	T
V\$PARAMETER	T/P
V\$BACKUP	T
V\$PX_PROCESS_SYSSTAT	T/P
V\$PROCESS	T
V\$WAITSTAT	T/P
V\$SYSTEM_EVENT	T/P

<u>Disk</u>	
V\$DATAFILE	T/P
V\$FILESTAT	T/P
V\$LOG	T
V\$LOG_HISTORY	T
V\$DBFILE	T/P
V\$TEMPFILE	P
V\$TEMPSTAT	P

<u>Memory</u>	
V\$BUFFER_POOL_STATISTICS	T/P
V\$DB_OBJECT_CACHE	T
V\$LIBRARYCACHE	P
V\$ROWCACHE	P
V\$SYSSTAT	T/P
V\$SGASTAT	P

<u>Contention</u>	
V\$LOCK	T/P
V\$ROLLNAME	T/P
V\$ROLLSTAT	T/P
V\$WAITSTAT	T/P
V\$LATCH	T/P

## Session-Related Statistics

<u>User/Session</u>	
V\$LOCK	P
V\$OPEN_CURSOR	T
V\$PROCESS	T
V\$SORT_USAGE	T/P
V\$SESSION	T/P
V\$SESSTAT	T/P
V\$TRANSACTION	T
V\$SESSION_EVENT	T/P
V\$SESSION_WAIT	T/P
V\$PX_SESSTAT	P
V\$PX_SESSION	P
V\$SESSION_OBJECT_CACHE	P

**T: Troubleshooting**

**T/P: Troubleshooting/Performance**



## Collecting Session-Related Statistics

